

Grado en Ingeniería Informática  
Curso 2017-2018

## *Trabajo Fin de Grado*

---

### Desarrollo de un entorno colaborativo en realidad virtual

---

Andrés Cepeda Fernández

Tutor: Telmo Zarraonandia Ayo

Leganés, junio de 2018



Esta obra se encuentra sujeta a la licencia Creative Commons **Reconocimiento – No Comercial – Sin Obra Derivada**



## RESUMEN

La tecnología de realidad virtual ha ganado en los últimos años una mayor importancia en el mercado, ya que cada vez más compañías ofrecen su alternativa para disfrutar de un entorno virtual. Esto ha provocado la aparición de dispositivos de realidad virtual económicamente asequibles para el usuario medio, lo cual ha aumentado su utilización. Por tanto, es el momento de explorar las oportunidades que ofrece de aplicación y mejora de actividades en distintos contextos.

Este trabajo se centra en los entornos colaborativos de visualización e interacción con mapas, tales como los centros coordinadores de emergencias. Estos centros suelen requerir pantallas de gran tamaño, de un coste generalmente elevado, para permitir el trabajo colaborativo del equipo. Se propone el empleo de la tecnología de realidad virtual como alternativa para abaratar los costes, además de permitir la colaboración entre miembros del equipo que estén situados en distintas localizaciones geográficas.

La propuesta consiste en desarrollar un entorno colaborativo en realidad virtual de visualización e interacción con mapas, que permita la colaboración con entornos sin realidad virtual. De esta forma, se ofrece la oportunidad de utilizar el sistema como complemento de la tecnología existente. Asimismo, permite estudiar la eficiencia de nuevas formas de interacción en este tipo de sistemas.

Para conseguir los objetivos descritos, se lleva a cabo un proyecto de desarrollo de software, en el cual se desarrolla una aplicación basada en realidad virtual en la que varios usuarios puedan comunicarse a través de la interacción con un mapa. Estos usuarios pueden añadir, eliminar y destacar marcadores, además de utilizar un chat de voz para comunicarse entre ellos.

Tras la realización de pruebas funcionales y el análisis de los objetivos planteados al inicio del proyecto, se puede concluir en que estos se han cumplido. No obstante, al tratarse de una primera versión, el resultado obtenido puede estar sujeto a cambios futuros, ya sea para ampliar sus funcionalidades o mejorar las ya existentes.



## **DEDICATORIA**

A mis padres, Antonio y Merche, por apoyarme en todo momento y enseñarme que, a base de esfuerzo, se puede conseguir lo que uno se proponga.

A mi novia, Tessa, porque gracias a ella he podido mantenerme en pie en los momentos más difíciles, aguantando lo inaguantable.

A los miembros de UC3M PALS: Diego, Edu, Hervás, Lute y Manu, porque gracias a ellos estos cuatro años han sido muchísimo más llevaderos.

Por último, y no por ello menos importante, a mi tutor, Telmo, por darme la oportunidad de hacer este TFG y, sobre todo, por estar siempre disponible cuando le he necesitado.

Gracias, simplemente gracias.

.



# ÍNDICE DE CONTENIDOS

1. INTRODUCCION .....	1
1.1. Contexto.....	1
1.2. Motivación del trabajo.....	2
1.3. Objetivos .....	3
1.4. Herramientas utilizadas .....	3
1.5. Estructura del documento.....	4
2. ENTORNO SOCIOECONÓMICO.....	6
2.1. Impacto socioeconómico .....	6
2.2. Presupuesto.....	8
3. MARCO REGULADOR .....	10
4. ESTADO DEL ARTE .....	12
4.1. Realidad virtual .....	12
4.1.1. Definición y conceptos básicos .....	12
4.1.2. Ventajas.....	14
4.1.3. Desventajas .....	15
4.1.4. Ejemplos de aplicaciones.....	16
4.2. Motores de juego .....	17
4.3. Dispositivos .....	18
4.3.1. Carcasas o cascos sin pantalla.....	18
4.3.2. Cascos con pantalla incorporada.....	19
4.4. Librerías .....	20
4.4.1. Realidad virtual .....	20
4.4.2. Interacción con mapas .....	21
4.4.3. Colaboración.....	21
5. ANÁLISIS .....	23
5.1. Descripción general de la aplicación .....	23
5.2. Requisitos de usuario .....	24
5.2.1. Capacidades generales.....	24
5.2.2. Restricciones generales .....	24
5.2.3. Características del usuario .....	24
5.2.4. Entorno operativo .....	25
5.2.5. Entorno de seguridad .....	25
5.2.6. Suposiciones y dependencias.....	25

5.2.7.	Requisitos específicos.....	26
5.3.	Casos de uso .....	39
5.4.	Requisitos de software .....	49
5.4.1.	Requisitos funcionales.....	51
5.4.2.	Requisitos de rendimiento.....	67
5.4.3.	Requisitos operacionales .....	67
5.4.4.	Requisitos de calidad .....	69
5.4.5.	Requisitos de interfaz .....	70
5.4.6.	Requisitos de usabilidad .....	71
5.5.	Análisis de clases .....	72
5.6.	Trazabilidad.....	76
5.6.1.	Requisitos de Usuario – Requisitos de Software.....	76
5.6.2.	Casos de Uso – Requisitos de Software .....	78
5.6.3.	Clases – Requisitos de Software .....	80
6.	DISEÑO.....	82
6.1.	Arquitectura.....	82
6.2.	Descripción de los componentes.....	84
6.3.	Trazabilidad Componentes – Requisitos de Software .....	90
6.4.	Elementos gráficos y controles de juego.....	92
6.4.1.	Elementos gráficos .....	92
6.4.2.	Controles de juego .....	95
7.	IMPLEMENTACIÓN .....	99
7.1.	Escenas .....	99
7.2.	Interacción con mapas y uso de marcadores .....	100
7.3.	Colaboración.....	101
8.	PRUEBAS .....	105
8.1.	Especificación del conjunto de pruebas.....	105
8.2.	Resultados de las pruebas .....	127
9.	CONCLUSIONES.....	131
9.1.	Objetivos cumplidos.....	131
9.2.	Futuras líneas de trabajo .....	132
10.	GESTION DEL PROYECTO .....	134
10.1.	Planificación .....	134
10.2.	Método de desarrollo .....	135
11.	BIBLIOGRAFÍA.....	137



12.	DEVELOPMENT OF A COLLABORATIVE VIRTUAL REALITY ENVIRONMENT .....	140
12.1.	Introduction.....	140
12.1.1.	Context.....	140
12.1.2.	Motivation of the project.....	141
12.1.3.	Objectives .....	142
12.2.	Methods.....	143
12.2.1.	Virtual reality .....	143
12.2.2.	State of the art .....	145
12.3.	Results.....	147
12.3.1.	Software development project .....	147
12.3.2.	General description of the developed application .....	147
12.3.3.	Testing results.....	148
12.4.	Conclusions.....	150
12.4.1.	Achieved objectives .....	150
12.4.2.	Future lines of work.....	150



## ÍNDICE DE FIGURAS

Fig. 4.1. Diagrama del continuo de virtualidad de Milgram y Kishino .....	12
Fig. 5.1. Diagrama de casos de uso .....	39
Fig. 5.2. Diagrama de clases.....	72
Fig. 6.1. Diagrama de componentes.....	83
Fig. 6.2. Texturas de marcadores .....	92
Fig. 6.3. Comparación entre un marcador destacado y otro sin destacar .....	93
Fig. 6.4. Textura base de las flechas .....	93
Fig. 6.5. Lista de usuarios .....	94
Fig. 6.6. Controles de juego en entorno sin realidad virtual .....	96
Fig. 6.7. Controles de juego en entorno con realidad virtual.....	98
Fig. 7.1. Escena inicial de configuración (izquierda) y escena principal (derecha) .....	99
Fig. 7.2. Cálculo de la rotación y la posición de las flechas .....	103
Fig. 10.1. Diagrama de Gantt inicial del proyecto.....	134
Fig. 10.2. Diagrama de Gantt final del proyecto .....	135
Fig. 10.3. Modelo en cascada con retroalimentación .....	136



## ÍNDICE DE TABLAS

TABLA 3.1: COSTE DE PERSONAL.....	8
TABLA 3.2: COSTE DE RECURSOS.....	9
TABLA 4.1: COMPARACIÓN ENTRE MOTORES DE JUEGO (Ref).....	17
TABLA 4.2: COMPARACIÓN ENTRE CASCOS DE REALIDAD VIRTUAL.....	19
TABLA 4.3: COMPARACIÓN ENTRE LIBRERÍAS DE COLABORACIÓN (Ref) .....	22
TABLA 5.1: PLANTILLA DE REQUISITOS DE USUARIO.....	27
TABLA 5.2: REQUISITO DE USUARIO RU-C01 .....	28
TABLA 5.3: REQUISITO DE USUARIO RU-C02 .....	28
TABLA 5.4: REQUISITO DE USUARIO RU-C03 .....	29
TABLA 5.5: REQUISITO DE USUARIO RU-C04 .....	29
TABLA 5.6: REQUISITO DE USUARIO RU-C05 .....	29
TABLA 5.7: REQUISITO DE USUARIO RU-C06 .....	30
TABLA 5.8: REQUISITO DE USUARIO RU-C07 .....	30
TABLA 5.9: REQUISITO DE USUARIO RU-C08 .....	30
TABLA 5.10: REQUISITO DE USUARIO RU-C09 .....	31
TABLA 5.11: REQUISITO DE USUARIO RU-C10 .....	31
TABLA 5.12: REQUISITO DE USUARIO RU-C11 .....	31
TABLA 5.13: REQUISITO DE USUARIO RU-C12 .....	32
TABLA 5.14: REQUISITO DE USUARIO RU-C13 .....	32
TABLA 5.15: REQUISITO DE USUARIO RU-C14 .....	32
TABLA 5.16: REQUISITO DE USUARIO RU-C15 .....	33
TABLA 5.17: REQUISITO DE USUARIO RU-C16 .....	33
TABLA 5.18: REQUISITO DE USUARIO RU-C17 .....	34
TABLA 5.19: REQUISITO DE USUARIO RU-R01 .....	34
TABLA 5.20: REQUISITO DE USUARIO RU-R02 .....	35
TABLA 5.21: REQUISITO DE USUARIO RU-R03 .....	35
TABLA 5.22: REQUISITO DE USUARIO RU-R04 .....	35
TABLA 5.23: REQUISITO DE USUARIO RU-R05 .....	36
TABLA 5.24: REQUISITO DE USUARIO RU-R06 .....	36
TABLA 5.25: REQUISITO DE USUARIO RU-R07 .....	36
TABLA 5.26: REQUISITO DE USUARIO RU-R08 .....	37

TABLA 5.27: REQUISITO DE USUARIO RU-R09 .....	37
TABLA 5.28: REQUISITO DE USUARIO RU-R10 .....	37
TABLA 5.29: REQUISITO DE USUARIO RU-R11 .....	38
TABLA 5.30: REQUISITO DE USUARIO RU-R12 .....	38
TABLA 5.31: PLANTILLA DE CASOS DE USO .....	40
TABLA 5.32: CASO DE USO CU-01 .....	41
TABLA 5.33: CASO DE USO CU-02 .....	42
TABLA 5.34: CASO DE USO CU-03 .....	43
TABLA 5.35: CASO DE USO CU-04 .....	43
TABLA 5.36: CASO DE USO CU-05 .....	44
TABLA 5.37: CASO DE USO CU-06 .....	45
TABLA 5.38: CASO DE USO CU-07 .....	46
TABLA 5.39: CASO DE USO CU-08 .....	47
TABLA 5.40: CASO DE USO CU-09 .....	48
TABLA 5.41: CASO DE USO CU-10 .....	48
TABLA 5.42: PLANTILLA DE REQUISITOS DE SOFTWARE .....	50
TABLA 5.43: REQUISITO DE SOFTWARE RS-F01 .....	51
TABLA 5.44: REQUISITO DE SOFTWARE RS-F02 .....	51
TABLA 5.45: REQUISITO DE SOFTWARE RS-F03 .....	51
TABLA 5.46: REQUISITO DE SOFTWARE RS-F04 .....	52
TABLA 5.47: REQUISITO DE SOFTWARE RS-F05 .....	52
TABLA 5.48: REQUISITO DE SOFTWARE RS-F06 .....	52
TABLA 5.49: REQUISITO DE SOFTWARE RS-F07 .....	53
TABLA 5.50: REQUISITO DE SOFTWARE RS-F08 .....	53
TABLA 5.51: REQUISITO DE SOFTWARE RS-F09 .....	53
TABLA 5.52: REQUISITO DE SOFTWARE RS-F10 .....	54
TABLA 5.53: REQUISITO DE SOFTWARE RS-F11 .....	54
TABLA 5.54: REQUISITO DE SOFTWARE RS-F12 .....	54
TABLA 5.55: REQUISITO DE SOFTWARE RS-F13 .....	55
TABLA 5.56: REQUISITO DE SOFTWARE RS-F14 .....	55
TABLA 5.57: REQUISITO DE SOFTWARE RS-F15 .....	55
TABLA 5.58: REQUISITO DE SOFTWARE RS-F16 .....	56
TABLA 5.59: REQUISITO DE SOFTWARE RS-F17 .....	56
TABLA 5.60: REQUISITO DE SOFTWARE RS-F18 .....	56

TABLA 5.61: REQUISITO DE SOFTWARE RS-F19.....	57
TABLA 5.62: REQUISITO DE SOFTWARE RS-F20.....	57
TABLA 5.63: REQUISITO DE SOFTWARE RS-F21.....	57
TABLA 5.64: REQUISITO DE SOFTWARE RS-F22.....	58
TABLA 5.65: REQUISITO DE SOFTWARE RS-F23.....	58
TABLA 5.66: REQUISITO DE SOFTWARE RS-F24.....	58
TABLA 5.67: REQUISITO DE SOFTWARE RS-F25.....	59
TABLA 5.68: REQUISITO DE SOFTWARE RS-F26.....	59
TABLA 5.69: REQUISITO DE SOFTWARE RS-F27.....	59
TABLA 5.70: REQUISITO DE SOFTWARE RS-F28.....	60
TABLA 5.71: REQUISITO DE SOFTWARE RS-F29.....	60
TABLA 5.72: REQUISITO DE SOFTWARE RS-F30.....	60
TABLA 5.73: REQUISITO DE SOFTWARE RS-F31.....	61
TABLA 5.74: REQUISITO DE SOFTWARE RS-F32.....	61
TABLA 5.75: REQUISITO DE SOFTWARE RS-F33.....	61
TABLA 5.76: REQUISITO DE SOFTWARE RS-F34.....	62
TABLA 5.77: REQUISITO DE SOFTWARE RS-F35.....	62
TABLA 5.78: REQUISITO DE SOFTWARE RS-F36.....	62
TABLA 5.79: REQUISITO DE SOFTWARE RS-F37.....	63
TABLA 5.80: REQUISITO DE SOFTWARE RS-F38.....	63
TABLA 5.81: REQUISITO DE SOFTWARE RS-F39.....	63
TABLA 5.82: REQUISITO DE SOFTWARE RS-F40.....	64
TABLA 5.83: REQUISITO DE SOFTWARE RS-F41.....	64
TABLA 5.84: REQUISITO DE SOFTWARE RS-F42.....	64
TABLA 5.85: REQUISITO DE SOFTWARE RS-F43.....	65
TABLA 5.86: REQUISITO DE SOFTWARE RS-F44.....	65
TABLA 5.87: REQUISITO DE SOFTWARE RS-F45.....	65
TABLA 5.88: REQUISITO DE SOFTWARE RS-F46.....	66
TABLA 5.89: REQUISITO DE SOFTWARE RS-F47.....	66
TABLA 5.90: REQUISITO DE SOFTWARE RS-F48.....	66
TABLA 5.91: REQUISITO DE SOFTWARE RS-RE01.....	67
TABLA 5.92: REQUISITO DE SOFTWARE RS-OP01.....	67
TABLA 5.93: REQUISITO DE SOFTWARE RS-OP02.....	67
TABLA 5.94: REQUISITO DE SOFTWARE RS-OP03.....	68

TABLA 5.95: REQUISITO DE SOFTWARE RS-OP04.....	68
TABLA 5.96: REQUISITO DE SOFTWARE RS-OP05.....	68
TABLA 5.97: REQUISITO DE SOFTWARE RS-OP06.....	69
TABLA 5.98: REQUISITO DE SOFTWARE RS-CA01 .....	69
TABLA 5.99: REQUISITO DE SOFTWARE RS-CA02.....	69
TABLA 5.100: REQUISITO DE SOFTWARE RS-IO1 .....	70
TABLA 5.101: REQUISITO DE SOFTWARE RS-IO2.....	70
TABLA 5.102: REQUISITO DE SOFTWARE RS-IO3.....	70
TABLA 5.103: REQUISITO DE SOFTWARE RS-U01 .....	71
TABLA 5.104: REQUISITO DE SOFTWARE RS-U02 .....	71
TABLA 5.105: PLANTILLA DE IDENTIFICACIÓN DE CLASES.....	73
TABLA 5.106: CLASE SALA.....	73
TABLA 5.107: CLASE USUARIO.....	74
TABLA 5.108: CLASE MARCADOR.....	74
TABLA 5.109: CLASE MAPA.....	75
TABLA 5.110: TRAZABILIDAD RU – RS (I) .....	76
TABLA 5.111: TRAZABILIDAD RU – RS (II) .....	77
TABLA 5.112: TRAZABILIDAD CU – RS (I) .....	78
TABLA 5.113: TRAZABILIDAD CU – RS (II) .....	79
TABLA 5.114: TRAZABILIDAD CLASES – RS (I) .....	80
TABLA 5.115: TRAZABILIDAD CLASES – RS (II).....	81
TABLA 6.1: PLANTILLA DE COMPONENTES .....	84
TABLA 6.2: COMPONENTE CO-01 .....	85
TABLA 6.3: COMPONENTE CO-02 .....	85
TABLA 6.4: COMPONENTE CO-03 .....	86
TABLA 6.5: COMPONENTE CO-04 .....	86
TABLA 6.6: COMPONENTE CO-05 .....	86
TABLA 6.7: COMPONENTE CO-06 .....	87
TABLA 6.8: COMPONENTE CO-07 .....	87
TABLA 6.9: COMPONENTE CO-08 .....	87
TABLA 6.10: COMPONENTE CO-09 .....	88
TABLA 6.11: COMPONENTE CO-10 .....	88
TABLA 6.12: COMPONENTE CO-11 .....	88
TABLA 6.13: COMPONENTE CO-12 .....	89



TABLA 6.14: COMPONENTE CO-13 .....	89
TABLA 6.15: COMPONENTE CO-14 .....	89
TABLA 6.16: TRAZABILIDAD CO – RS (I) .....	90
TABLA 6.17: TRAZABILIDAD CO – RS (II) .....	91
TABLA 6.18: CONTROLES DE JUEGO EN ENTORNO SIN REALIDAD VIRTUAL ...	95
TABLA 6.19: CONTROLES DE JUEGO EN ENTORNO CON REALIDAD VIRTUAL .	97
TABLA 8.1: PLANTILLA DE CASOS DE PRUEBA .....	106
TABLA 8.2: CASO DE PRUEBA CP-01 .....	106
TABLA 8.3: CASO DE PRUEBA CP-02 .....	107
TABLA 8.4: CASO DE PRUEBA CP-03 .....	107
TABLA 8.5: CASO DE PRUEBA CP-04 .....	108
TABLA 8.6: CASO DE PRUEBA CP-05 .....	108
TABLA 8.7: CASO DE PRUEBA CP-06 .....	109
TABLA 8.8: CASO DE PRUEBA CP-07 .....	109
TABLA 8.9: CASO DE PRUEBA CP-08 .....	110
TABLA 8.10: CASO DE PRUEBA CP-09 .....	110
TABLA 8.11: CASO DE PRUEBA CP-10 .....	111
TABLA 8.12: CASO DE PRUEBA CP-11 .....	111
TABLA 8.13: CASO DE PRUEBA CP-12 .....	112
TABLA 8.14: CASO DE PRUEBA CP-13 .....	112
TABLA 8.15: CASO DE PRUEBA CP-14 .....	113
TABLA 8.16: CASO DE PRUEBA CP-15 .....	113
TABLA 8.17: CASO DE PRUEBA CP-16 .....	114
TABLA 8.18: CASO DE PRUEBA CP-17 .....	114
TABLA 8.19: CASO DE PRUEBA CP-18 .....	115
TABLA 8.20: CASO DE PRUEBA CP-19 .....	115
TABLA 8.21: CASO DE PRUEBA CP-20 .....	116
TABLA 8.22: CASO DE PRUEBA CP-21 .....	116
TABLA 8.23: CASO DE PRUEBA CP-22 .....	117
TABLA 8.24: CASO DE PRUEBA CP-23 .....	117
TABLA 8.25: CASO DE PRUEBA CP-24 .....	118
TABLA 8.26: CASO DE PRUEBA CP-25 .....	118
TABLA 8.27: CASO DE PRUEBA CP-26 .....	119
TABLA 8.28: CASO DE PRUEBA CP-27 .....	119

TABLA 8.29: CASO DE PRUEBA CP-28 .....	120
TABLA 8.30: CASO DE PRUEBA CP-29 .....	120
TABLA 8.31: CASO DE PRUEBA CP-30 .....	121
TABLA 8.32: CASO DE PRUEBA CP-31 .....	121
TABLA 8.33: CASO DE PRUEBA CP-32 .....	122
TABLA 8.34: CASO DE PRUEBA CP-33 .....	122
TABLA 8.35: CASO DE PRUEBA CP-34 .....	123
TABLA 8.36: CASO DE PRUEBA CP-35 .....	124
TABLA 8.37: CASO DE PRUEBA CP-36 .....	125
TABLA 8.38: CASO DE PRUEBA CP-37 .....	126
TABLA 8.39: CASO DE PRUEBA CP-38 .....	126
TABLA 8.40: RESULTADOS DE LAS PRUEBAS (I) .....	127
TABLA 8.41: RESULTADOS DE LAS PRUEBAS (II) .....	128
TABLA 8.42: RESULTADOS DE LAS PRUEBAS (III) .....	129

# 1. INTRODUCCION

## 1.1. Contexto

A lo largo de los últimos años, se ha producido un gran desarrollo tecnológico, el cual ha propiciado la aparición de una gran cantidad de productos sobre los que se generan entornos virtuales. En caso de que el entorno sea completamente virtual, se habla de Realidad Virtual (VR).

Este proyecto se centra en el desarrollo en realidad virtual. Se trata de una experiencia en la que el usuario se encuentra inmerso en un entorno generado por ordenador. Esta es la principal diferencia con la realidad aumentada, la cual añade elementos virtuales a un entorno real.

El componente principal para la generación de estos entornos virtuales son unos cascos o gafas, también conocidos como *Head-Mounted Displays* o HMD. Estas gafas pueden consistir en un soporte para introducir un *smartphone* o, en el caso de los más sofisticados, poseer una pantalla propia. En el caso de la realidad aumentada, debido a la virtualización de una parte del entorno, es frecuente la utilización del *smartphone* directamente en lugar de colocarlo en un soporte. No obstante, existen modelos de gafas de realidad aumentada con pantalla propia como, por ejemplo, las HoloLens de Microsoft.

Actualmente, el uso de gafas con pantalla incorporada es mayor en realidad virtual que en el caso de la realidad aumentada. Esto se debe a que existe un menor número de HMD de realidad aumentada, y por el momento suelen ser versiones para desarrolladores. Por ello, su precio es considerablemente mayor que el de las gafas de realidad virtual, con un mayor número de versiones para consumidores.

En el caso de gafas con pantalla incorporada, cabe destacar que están comenzando a ser económicamente posibles para el usuario medio. Se trata de un área que ha ido ganando progresivamente una mayor importancia en el mercado, ya que cada vez más compañías ofrecen su alternativa para disfrutar de un entorno virtual. Se ha estimado que en 2018 se alcanzarían los 28 millones de usuarios de realidad virtual [1].

Es un hecho que uno de los principales usos que se le está dando a esta tecnología está asociado a la industria de los videojuegos. No obstante, también existen usos en educación [2], en el ámbito terapéutico [3] y el entrenamiento en situaciones de alto riesgo [4], entre otros.

Es importante aprovechar las posibilidades que pueden ofrecer las nuevas tecnologías, para así poder utilizarlas cuando sean más asequibles. Por ello, se debe experimentar con nuevos paradigmas de interacción, para así facilitar el desempeño de tareas en todas las áreas posibles. Se trata de una tecnología que todavía tiene margen de mejora, por lo que puede ofrecer infinitas posibilidades de cara al futuro.

## 1.2. Motivación del trabajo

La motivación principal de la realización de este trabajo es explorar las posibilidades que la tecnología de realidad virtual puede ofrecer en entornos colaborativos, en este caso, en los entornos de interacción y visualización con mapas.

La tecnología de realidad virtual ha ido ganando progresivamente una mayor importancia en el mercado, ya que cada vez más compañías ofrecen su alternativa para disfrutar de un entorno virtual. Esto ha provocado la aparición de dispositivos de realidad virtual a precio reducido, asequibles para el usuario medio. Aunque no se trata de una tecnología nueva, es ahora cuando está aumentando su utilización, al facilitar la adquisición de sus dispositivos como un objeto de uso doméstico o, al menos, fuera del ámbito de los laboratorios o las grandes empresas. Por tanto, es el momento de explorar las oportunidades que ofrece la aplicación de esta tecnología, así como la mejora de actividades en distintos contextos.

Este trabajo se centra en utilizar la tecnología de realidad virtual para ofrecer una alternativa más económica en los entornos colaborativos de visualización e interacción con mapas, tales como los centros coordinadores de emergencias. Estos centros suelen requerir pantallas de gran tamaño, de un coste generalmente elevado, para permitir el trabajo colaborativo del equipo sobre un mapa. Se propone el empleo de la tecnología de realidad virtual como alternativa a estas pantallas para abaratar los costes, además de permitir la colaboración entre miembros del equipo que estén situados en distintas localizaciones geográficas.

Entre las ventajas que ofrece la realidad virtual para la visualización e interacción con mapas, destaca el amplio campo de visión que ofrece al usuario, así como la posibilidad de crear entornos virtuales de gran extensión con una interacción realizada desde un espacio físico reducido. Esto permite disponer de un amplio lienzo sobre el que disponer la información, sin limitaciones a nivel espacial en el mundo virtual. Además, el usuario se encuentra inmerso en el entorno virtual, sin distracciones, lo cual permite facilitar y agilizar la realización de las tareas realizadas. Por ello, es importante destacar, además del impacto potencial a nivel económico, la utilidad que se puede ofrecer en el ámbito empresarial con la realización de este proyecto.

Por otra parte, este trabajo ofrecía la posibilidad de desarrollar con una tecnología novedosa con una gran proyección de futuro como es la realidad virtual. Cabe destacar que no se había trabajado con esta tecnología a lo largo del Grado, ni se había utilizado Unity como plataforma de desarrollo. Por esta razón, el trabajo requería un mayor esfuerzo inicial, ya que es necesario aprender a utilizar una nueva herramienta y adaptarse a esta nueva tecnología. No obstante, este proyecto se presentaba como una oportunidad para adquirir nuevas aptitudes, así como aplicar los conceptos aprendidos en los últimos años. Se trata de una preparación adicional que puede ser positiva de cara al futuro.

### 1.3. Objetivos

Los objetivos de este proyecto son los siguientes:

- **Desarrollo de un entorno colaborativo de visualización de mapas en realidad virtual.** Se trata del objetivo principal del proyecto. Varios usuarios se encuentran inmersos simultáneamente en un entorno virtual en el que se muestra un mapa. A través de la interacción con dicho mapa, serán capaces de comunicarse entre ellos. Se propone una alternativa que requiere menor coste y espacio físico que las pantallas de gran tamaño utilizadas actualmente para la interacción con mapas en centros de planificación de emergencias.
- **Extensión de la colaboración a entornos que no sean de realidad virtual.** Un usuario puede participar en la colaboración, aunque no posea gafas de realidad virtual. En este caso, se propone la utilización de un monitor corriente y una interacción con el mapa a través de ratón y teclado. Por tanto, un usuario que utilice las gafas de realidad virtual y otro que haga uso de un monitor para la visualización del mapa podrán comunicarse entre sí. De esta forma, se ofrece la oportunidad de utilizar un sistema basado en realidad virtual como complemento de la tecnología existente, y no como un reemplazo la misma.
- **Análisis de formas de interacción efectivas.** A pesar de no ser el objetivo principal del trabajo, el desarrollo del proyecto permitirá el estudio de nuevas formas de interacción aplicables a sistemas de visualización e interacción con mapas. Debido a que la realidad virtual se trata de una tecnología en la que se está comenzando a desarrollar, apenas existe información al respecto. La aplicación desarrollada será útil para comprobar la efectividad de las distintas alternativas que ofrece la realidad virtual para la interacción en este tipo de sistemas.

### 1.4. Herramientas utilizadas

Las herramientas que se han utilizado para llevar a cabo el desarrollo del proyecto son las siguientes:

- **Ordenador portátil.** Es el ordenador personal con el que se ha realizado la mayor parte del desarrollo. Además, se utilizaba a la hora de realizar las pruebas en un entorno no virtual. Se trata de un MSI GE62 Apache Pro, con un procesador Intel Core i7-7700HQ a 2,80GHz, una memoria de 16GB de tipo DDR4 y una tarjeta gráfica GTX 1050 Ti de 4GB. El sistema operativo que incluye es Microsoft Windows 10 Home.
- **Ordenador de sobremesa.** Debido a que el ordenador portátil no poseía los requisitos básicos para soportar el uso de las gafas de realidad virtual, se utilizó otro ordenador para desarrollar y realizar las pruebas de los componentes exclusivos de realidad virtual. Se trata de un ordenador con una placa base de Gigabyte, con un procesador Intel Core i5-6600 a 3,30GHz, una memoria RAM de 16GB y una tarjeta gráfica AMD Radeon 480 de 8GB. El sistema operativo que incluye es Microsoft Windows 10 Pro.

- **Gafas de realidad virtual.** En este caso, se ha utilizado un casco de realidad virtual de la compañía Oculus VR, en concreto el modelo Oculus Rift CV1. Se trata de un HMD pensado para consumidores.
- **Plataforma de desarrollo.** El desarrollo del proyecto se ha llevado a cabo en Unity 3D, en concreto la versión 5.4.0f3 para Windows 10. Se han utilizado diferentes SDK en la implementación, entre las que destacan Online Maps, Virtual Reality Toolkit y Photon Unity Networking.
- **Copias de seguridad.** El proyecto se ha ido subiendo regularmente a la plataforma Google Drive, comprimido en un archivo .zip. De esta forma, se han almacenado todas las versiones realizadas en el desarrollo.
- **Seguimiento del trabajo realizado a lo largo del proyecto.** Se ha ido anotando a lo largo del proyecto cada tarea realizada y las horas necesarias para su realización en un documento de Microsoft Excel. Esto permite tener un mayor control sobre todas las versiones que se subían en Google Drive, además de su utilidad para la gestión del proyecto.
- **Procesador de textos.** Para la realización de este documento, se ha utilizado Microsoft Word debido a su versatilidad en cuanto a formato, además de poseer experiencias previas con el uso de esta herramienta.

## 1.5. Estructura del documento

Los contenidos del documento se encuentran estructurados en torno a once capítulos:

1. **Introducción.** Contiene el contexto, la motivación y los objetivos planteados a la hora de realizar el trabajo. Se expone un problema, y se propone la solución que se va a desarrollar a lo largo del documento, así como las herramientas utilizadas para su consecución. Después, se describe la organización del documento. Por último, se incluye una sección con las abreviaturas, acrónimos y siglas utilizados a lo largo del trabajo.
2. **Entorno socioeconómico.** Se analiza el impacto socioeconómico esperado de la aplicación del resultado del proyecto, y se incluye el presupuesto de elaboración del TFG.
3. **Marco regulador.** Se realiza un análisis del marco regulador asociado al trabajo, en el que se detalla la legislación aplicable sobre la implementación descrita en el trabajo, así como los estándares técnicos aplicables.
4. **Estado del arte.** En este capítulo se revisarán las distintas tecnologías relacionadas con el desarrollo del proyecto. Se realiza un análisis de las alternativas existentes, para elegir aquellas que más se ajusten a las necesidades del proyecto.
5. **Análisis.** Se especifica cuál va a ser el funcionamiento del sistema a desarrollar, es decir, se realiza una especificación precisa de sus funcionalidades y restricciones. En primer lugar, se realiza una descripción general de la aplicación a desarrollar. Después, se detallará la especificación de requisitos de usuario, a

partir de la cual se analizarán los posibles casos de uso. Tras lo anterior, se generará la especificación técnica de requisitos de software y el análisis de clases. Por último, se creará un conjunto de matrices de trazabilidad para comprobar la consistencia de todos estos elementos.

6. **Diseño.** En este capítulo se define la arquitectura del sistema, dividiéndolo en subsistemas. Cada subsistema se dividirá en un conjunto de componentes, de los cuales se detallará su función específica.
7. **Implementación.** Se tratan los aspectos más importantes de la implementación de la aplicación. Se detalla el planteamiento para resolver cada una de las necesidades que se resuelven.
8. **Pruebas.** Se especificará una batería de pruebas que verifique el correcto funcionamiento de la aplicación, y se realizarán las pruebas especificadas, anotando los resultados obtenidos.
9. **Conclusiones.** Se verificará si los objetivos planteados al inicio del proyecto fueron alcanzados. Asimismo, se plantean las futuras líneas de trabajo tras conocer el resultado obtenido.
10. **Gestión del proyecto.** Se detalla la estimación del esfuerzo requerido para llevar a cabo el proyecto, para después realizar una comparación con el desempeño real en el mismo. Además, se incluye el método de desarrollo utilizado.
11. **Bibliografía.** Contiene una lista de las referencias bibliográficas empleadas a lo largo de todo el proyecto. En caso de encontrarse disponibles en internet, se incluirán los enlaces a las fuentes correspondientes.
12. **Development of a collaborative virtual reality environment.** Resumen del trabajo en el idioma inglés que incluye la introducción, objetivos, resultados, y conclusiones del mismo.

## **2. ENTORNO SOCIOECONÓMICO**

### **2.1. Impacto socioeconómico**

En primer lugar, se analizan las posibles aplicaciones en las que se puede asociar a este proyecto. En este caso, puede afectar a todo aquel sector en el que se realice una interacción y visualización de mapas de forma colectiva. La principal aplicación en la que se basa este proyecto es su utilización en centros coordinadores de emergencias, en los cuales se precisa de una comunicación constante y una frecuente utilización de mapas para la realización de sus tareas. No obstante, este proyecto también puede tener utilidad para la meteorología, en cuanto al análisis colectivo de mapas meteorológicos, o para la planificación de redes de comunicaciones, tales como redes de carreteras, vías de comunicación o vías férreas.

Este proyecto se muestra como una alternativa más económica a la utilización de pantallas de gran tamaño en entornos de la interacción con mapas de forma colaborativa. De esta forma, la inversión necesaria en cuanto a hardware será potencialmente menor. A modo de ejemplo, el presupuesto asociado a la adquisición de estas pantallas sería de alrededor de 60.000 euros, mientras que la compra de cada equipo que soporte realidad virtual es de aproximadamente 2.000 euros, teniendo en cuenta el coste de un casco de realidad virtual, así como de un ordenador que posea una tarjeta gráfica lo suficientemente potente. Considerando estos datos, el presupuesto de una pantalla sería equivalente a 30 equipos de realidad virtual.

No es necesario sustituir la tecnología ya existente, sino que se puede emplear como un complemento de la misma. De este modo, no se obliga a desechar las pantallas que se utilizan actualmente, lo cual supondría un desaprovechamiento de una inversión anterior. Debido a la posible compatibilidad con tecnologías anteriores, estas se podrán mantener hasta encontrar otro uso o hasta finalizar su vida útil.

El potencial impacto a nivel económico viene acompañado de la utilidad práctica de la solución propuesta en relación con las soluciones anteriores:

- Basado en un entorno inmersivo, sin distracciones, que puede facilitar y agilizar la realización de las tareas correspondientes. Este hecho puede implicar una mayor eficiencia en el entorno de trabajo, a excepción de un periodo inicial en el que los empleados tendrán que adaptarse al uso de los nuevos dispositivos. Este periodo será cada vez menor por el crecimiento del número de usuarios de realidad virtual en los hogares.
- Dos usuarios pueden interactuar entre ellos, independientemente de su situación geográfica. Este hecho puede reducir la movilidad, lo cual implica un ahorro en términos de tiempo y dinero.
- Posibilidad de crear diferentes entornos virtuales para cada grupo de usuarios. En cada uno de ellos, se podría utilizar un mapa distinto y cobijar a diferentes usuarios, de tal forma que no se produzcan interferencias entre las comunicaciones de cada grupo de usuarios.



- Portabilidad de los equipos y mayor facilidad en su sustitución en caso de avería.
- Reducción del espacio físico que requieren los equipos en relación con tamaño de las pantallas de gran tamaño.

En lo relativo a cuestiones sociales, cabe destacar posibilidad de reducir la movilidad anteriormente mencionada. Debido a que es necesario realizar un control sobre las emergencias ocurridas a lo largo de todos los países del mundo, este proyecto podría hablarse de una extensión geográfica mundial.

El número de personas o grupos afectados por la implementación de este proyecto es muy específico, dado que se trata de una solución aplicable únicamente a entornos de visualización e interacción con mapas. En cuanto al grado de incertidumbre del impacto, depende de cuestiones sociales, en concreto de la aceptación de nuevas tecnologías como la realidad virtual en el entorno laboral. Poco a poco va aumentando el uso de los dispositivos de realidad virtual, debido a que es posible adquirirlos a un precio asequible (para el usuario medio), y periodo de adaptación se puede producir en los hogares. No obstante, el uso que se da en estos casos está relacionado con el entretenimiento, por lo que se trata de un bien prescindible.

También cabe destacar la incertidumbre en cuanto a los efectos de una exposición prolongada de un casco de realidad virtual. No se considera un cambio en cuanto al número de empleados, ya que el número de personas necesarias para realizar las tareas de interacción y visualización de mapas será el mismo. No obstante, debido esta la incertidumbre acerca de los efectos que puede producir la exposición prolongada, podrá existir una restricción para su uso. Este aspecto podría implicar la realización de otras tareas por parte del mismo empleado.

En cuanto al plan de explotación del proyecto, cabe destacar que va a ser publicado en el repositorio institucional de la Universidad Carlos III de Madrid, por lo que se fomenta su difusión. Además, se realizarán experimentos posteriores para verificar la utilidad del proyecto, así como para encontrar posibles mejoras a realizar. Esto quedará reflejado en un artículo científico posterior a la realización de este proyecto, con el objetivo de la divulgación de los resultados obtenidos.

En resumen, el análisis del impacto socioeconómico ha arrojado resultados positivos en relación con el impacto económico, y un posible impacto social que se puede considerar despreciable. Se concluye en un potencial impacto económico positivo debido a reducción de costes asociados a los equipos necesarios para la realización de las diferentes tareas. Aunque el proyecto tenga aplicaciones en sectores muy específicos, se trata de sectores existentes en cualquier nación, por lo que es posible una extensión geográfica a nivel mundial. No obstante, existe un grado de incertidumbre debido a que requiere una previa aceptación de la tecnología por parte de los sectores implicados, teniendo en cuenta que estos valorarán los posibles problemas que, de momento, existen asociados a la realidad virtual. Estos problemas fueron descritos en el capítulo asociado al marco regulador del proyecto.

## 2.2. Presupuesto

El presupuesto del proyecto se ha calculado sumando el coste asociado al personal y el relativo a los recursos utilizados.

En el caso del coste del personal, se definen una serie de roles que se han ido adoptando a lo largo del proyecto. Cada rol estará asociado a un conjunto de tareas y tendrá un salario por hora específico.

- **Jefe de proyecto.** Se responsabiliza de planificar, ejecutar y monitorizar las tareas llevadas a cabo en la elaboración del sistema.
- **Analista de sistemas.** Su función es realizar el análisis de requisitos, así como analizar la estructura básica de la aplicación y generar los diagramas de alto nivel del sistema.
- **Arquitecto software.** Genera el diseño de la arquitectura del sistema, y se encarga de que el producto final se ajuste al diseño realizado.
- **Desarrollador.** Se encarga de la codificación e implementación del sistema, tras las pautas definidas por los procesos de análisis y diseño.
- **Responsable de pruebas.** Su cometido es definir y ejecutar un conjunto de pruebas funcionales sobre la aplicación desarrollada, con el objetivo de verificar su correcto funcionamiento.

La tabla 3.1 representa el coste asociado al personal, basado en el número de horas dedicadas a lo largo del proyecto en los distintos roles descritos.

TABLA 2.1: COSTE DE PERSONAL

Rol	Salario/h (€)	Tiempo dedicado (h)	Coste total (€)
Jefe de Proyecto	20,00	55	1.100,00
Analista de sistemas	15,80	50	790,00
Arquitecto software	18,75	35	656,25
Desarrollador	13,48	115	1.550,20
Responsable de pruebas	12,50	20	250,00
<b>Total</b>		<b>275</b>	<b>4.346,45</b>

En el caso de los recursos utilizados, se tendrá en cuenta el coste del hardware y el software requerido a lo largo de todo el proyecto, teniendo en cuenta el periodo de

tiempo en el que ha sido utilizado y su vida útil en el caso de hardware, y el coste de la licencia correspondiente en el caso del software.

TABLA 2.2: COSTE DE RECURSOS

Recurso	Precio (€)	Vida útil (meses)	Uso (meses)	Coste total (€)
MSI GE62 Apache Pro	1200,00	36	7	233,33
Ordenador Gigabyte	2500,00	48	6	312,50
Oculus Rift CV1	600,00	24	6	150,00
			<b>Total</b>	<b>695,83</b>

El presupuesto del proyecto, teniendo en cuenta el coste de personal y el de los recursos utilizados, asciende a un total de 5.042,28 €.

### 3. MARCO REGULADOR

En este apartado, se realiza un análisis de la legislación y de los estándares aplicables a la implementación descrita en el trabajo. Se tendrán en cuenta los problemas asociados de la tecnología utilizada, en este caso, la realidad virtual.

La mayor parte de los problemas de la realidad virtual están asociados a aspectos de salud y seguridad:

- Su uso puede provocar síntomas relacionados con el mareo por movimiento: visión alterada, desorientación, alteraciones en el equilibrio graves mareos, convulsiones, espasmos ópticos o musculares, movimientos involuntarios [5].
- El peso del HMD y a la utilización de mandos pueden producir dolores en músculos, articulaciones, cuello, manos o piel [5].
- Posibilidad de transmisión de enfermedades contagiosas al compartir el equipo entre varias personas [5].
- El usuario se encuentra inmerso en un entorno virtual, por lo que su uso prolongado puede producir problemas psicológicos desde distorsión de la realidad, hasta disparar o agravar desórdenes emocionales.
- Peligro de golpearse con objetos situados en la misma habitación y causar daños físicos graves.
- El usuario no es consciente de lo que sucede en el mundo real, debido a una inmersión en un entorno virtual. Esto puede generar dificultades en el caso de incendios, derrumbamientos, robos...

Teniendo en cuenta los problemas mencionados, y que el uso del proyecto a desarrollar está destinado al ámbito profesional, en el que puede darse un uso diario de un HMD, se proponen ciertas consideraciones para tener en cuenta.

- Para evitar que los empleados sufran de daño físico o psicológico debido a un uso excesivo de estos dispositivos, se sugiere la obligatoriedad de un control del tiempo dedicado por cada empleado en el que se tengan en cuenta sus características físicas y psicológicas. Además, se deberán respetar las recomendaciones de uso del dispositivo.
- En el caso de accidentes ocasionados en el entorno real, se deberá restringir la utilización de dispositivos de realidad virtual a entornos cerrados que no contengan ningún obstáculo. Asimismo, deberá existir personal dedicado a la seguridad de los empleados que estén trabajando en entornos virtuales.

En cuanto a los estándares técnicos, debido a la novedad de la tecnología, no han comenzado a plantearse hasta finales de 2016. En diciembre de 2016, se crea la iniciativa de Khronos Group [6], que trata de definir un estándar abierto, tanto para realidad virtual, como para realidad aumentada. Este estándar tiene como nombre OpenXR [7], y surge a raíz de la creciente diversidad en motores de juego, aplicaciones y dispositivos de realidad virtual y aumentada. Esta diversidad implica que cada plataforma pueda utilizar únicamente los dispositivos con los que existe una

compatibilidad predefinida. Esto implica un aumento en el coste de desarrollo de las aplicaciones.

La solución a este problema es la creación de un estándar a través del cual toda plataforma pueda ser compatible con cualquier dispositivo, debido a la utilización de una arquitectura común.

Actualmente, OpenXR se encuentra en fase de desarrollo, pero ya cuenta con el apoyo de compañías líderes en realidad virtual y aumentada como Oculus, Samsung o HTC, entre otros. La consolidación de este estándar favorecerá la implementación de las aplicaciones, ya que evitará el esfuerzo que supone la compatibilidad con diversos dispositivos, y los desarrolladores podrán centrarse en la innovación de sus productos.

## 4. ESTADO DEL ARTE

En este capítulo se revisarán las distintas tecnologías relacionadas con el desarrollo del proyecto. Se realiza un análisis de las alternativas existentes, tanto hardware como software, que pueden utilizarse para resolver el problema planteado. Después, se utiliza la información recabada para tomar las decisiones que más se ajusten a las necesidades del proyecto y los recursos disponibles.

### 4.1. Realidad virtual

Antes de iniciar el análisis del estado del arte, se hará una breve introducción de la tecnología de realidad virtual. Se define la tecnología y una serie de conceptos básicos, así como ventajas, desventajas y ejemplos de aplicaciones.

#### 4.1.1. Definición y conceptos básicos

Tal y como se muestra en la figura 4.1, se realiza una distinción entre el mundo real y el mundo virtual, el cual es completamente generado por computadores. El espectro comprendido entre el mundo real y mundo virtual, en el que se fusionan ambos mundos, se encuentra la denominada Realidad Mixta. Dependiendo de la proporción de cada uno de estos mundos, se distingue entre Realidad Aumentada y Virtualidad Aumentada. El presente trabajo se centra en la Realidad Virtual (VR), en la que el mundo es completamente virtual.

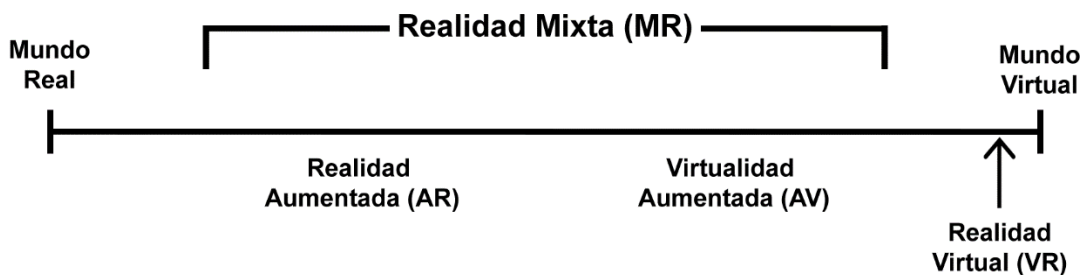


Fig. 4.1. Diagrama del continuo de virtualidad de Milgram y Kishino

La realidad virtual es un tipo de realidad mediada por computador que consiste en la percepción de un entorno totalmente virtual como el mundo real, es decir, es la inmersión del usuario en una realidad generada por computador.

Dos propiedades principales de esta tecnología son la inmersión y la presencia. Se trata de dos conceptos que se suelen confundir, por lo que es necesario conocer la diferencia entre ellos. En [8], se define la inmersión como el nivel objetivo de fidelidad sensorial que proporciona un sistema de realidad virtual, mientras que la presencia está relacionada con la respuesta psicológica del usuario en cada momento. De esta forma, la inmersión se puede determinar a partir del software de renderizado, así como el tipo de pantalla utilizada. En cambio, la presencia es una característica subjetiva e individual,

que depende tanto del contexto como del usuario, y cuyo nivel cambia con el paso del tiempo.

Mel Slater [9] utiliza varias analogías para explicar la diferencia entre ambos conceptos. En una de ellas, la distinción la realiza basándose en la ciencia del color. Por un lado, asocia la inmersión a la descripción objetiva de los colores mediante su longitud de onda, en la que dos colores con la misma longitud de onda serán iguales, independientemente de su contexto. Por otro lado, la presencia sería la percepción del ser humano de los colores ya que, no sólo se pueden percibir dos longitudes de onda como el mismo color, sino también el mismo color puede ser percibido diferente dependiendo de la persona que lo esté observando. Por ello, se concibe la presencia como la percepción de la inmersión.

Por lo comentado anteriormente, tanto la inmersión como la presencia son imprescindibles para una buena experiencia de usuario. Cuanto mejor sea el hardware y el software utilizado, mayor será la inmersión del usuario. Asimismo, una mayor inmersión afectará positivamente a la sensación de presencia. Uno de los objetivos principales al desarrollar una aplicación en realidad virtual es alcanzar la idea de presencia.

Hoy en día, los entornos inmersivos se generan utilizando HMDs o gafas de realidad virtual. Estos dispositivos se colocan en la cabeza, de tal forma que la pantalla se sitúe enfrente de los ojos y se observe el mundo virtual en todo momento. Asimismo, el HMD tendrá control de la posición y la rotación de la cabeza gracias a la utilización de sensores externos o incorporados.

Otro concepto importante es el de *room scale*. Consiste en la utilización del rastreo de movimiento del HMD para simular el movimiento en el mundo real dentro de un área de juego. En este caso, es necesario que el entorno real se encuentre vacío y sin obstáculos, de tal forma que se eviten accidentes.

En cuanto a las formas de interacción utilizadas:

- **Mandos que simulen las manos en el entorno virtual.** Estos mandos contienen sensores en los botones que detectan la proximidad a estos, de tal forma que se simule el movimiento de los dedos de la mano, consiguiendo una interacción natural. Ejemplos de estos mandos son Oculus Touch o los que proporciona HTC Vive.
- **Leap motion.** Otra forma de interacción natural que consiste en la detección del movimiento de las manos para realizar las distintas acciones.
- **Utilización de rayos.** Estos pueden proceder del centro de la mirada (*gaze-input*) o de las manos (Oculus Touch). Esto permite seleccionar objetos lejanos y manipularlos, o la navegación por el entorno virtual, teletransportando al usuario al lugar que indica el rayo.
- **Gamepads.** Son mandos de videojuegos que proporcionan *joysticks*, a través de los cuales se produce el desplazamiento y el giro de la cámara, y poseen botones, con los que se pueden realizar las distintas acciones. Un ejemplo de ello es la posibilidad de utilizar el mando de Xbox en Oculus Rift.

Actualmente, uno de los usos más extendidos aplicaciones de realidad virtual proceden de la industria de los videojuegos, pero se están explorando nuevas posibilidades que la realidad virtual puede ofrecer en otros ámbitos. Ejemplos de ello son el tratamiento de fobias [10] o las visitas virtuales a réplicas de museos o monumentos históricos [11]. Otro uso frecuente es el soporte de simulaciones para entrenamientos, con el objetivo de estimular el aprendizaje y/o eliminar los riesgos asociados a su actividad real [12].

#### 4.1.2. Ventajas

Entre las ventajas de la tecnología de realidad virtual, destacan:

- **Inmersión y presencia.** Se trata de la única tecnología que puede dar la sensación de que se está en un lugar diferente, y puede llegar al extremo en que se produzca una inmersión total, en la que el usuario crea que realmente se encuentra físicamente en un entorno virtual.
- **Simulación.** Permiten la práctica en un entorno totalmente controlado y sin riesgos.
- **Ahorro en costes.** Se puede recrear cualquier espacio interactivo ahorrando en costes de generación física del entorno.
- **Entretenimiento.** El uso principal de esta tecnología está asociada al ocio, debido a su corte audiovisual. Aporta nuevas experiencias al usuario, imposibles de disfrutar con otra tecnología.

Existen otras ventajas que deben tomarse en consideración por este proyecto en específico, las cuales fueron descritas en el análisis del impacto socioeconómico:

- **Eficiencia en las tareas.** La realidad virtual está basada en un entorno inmersivo, sin distracciones, que puede facilitar y agilizar la realización de las tareas correspondientes. Este hecho puede implicar una mayor eficiencia en el entorno de trabajo, a excepción de un periodo inicial en el que los empleados tendrán que adaptarse al uso de los nuevos dispositivos. Este periodo será cada vez menor por el crecimiento del número de usuarios de realidad virtual en los hogares.
- **Grandes entornos virtuales en espacios físicos reducidos.** No existen limitaciones espaciales en los mundos virtuales, por lo que se puede presentar una cantidad considerable de información. Se precisa únicamente de un espacio reducido para ejecutar aplicaciones donde no es necesario el desplazamiento físico del usuario, como es el caso de la interacción con mapas. Esto implica una reducción del espacio físico que requieren los equipos en relación con tamaño de las pantallas de gran tamaño.
- **Portabilidad de los equipos.** Debido a su tamaño y a la modularidad de la solución, los equipos son fácilmente transportables y sustituibles en caso de avería.



- **Diferentes situaciones geográficas.** Dos usuarios pueden interactuar entre ellos, independientemente de su situación geográfica. Este hecho puede reducir la movilidad, lo cual implica un ahorro en términos de tiempo y dinero.
- **Creación de varios entornos virtuales.** Posibilidad de crear diferentes entornos virtuales para cada grupo de usuarios. En cada uno de ellos, se podría utilizar un mapa distinto y cobijar a diferentes usuarios, de tal forma que no se produzcan interferencias entre las comunicaciones de cada grupo de usuarios, sin impacto en los costes.

#### 4.1.3. Desventajas

Tal y como se definió en el marco regulador del proyecto, las desventajas de la realidad virtual son:

- Su uso puede provocar síntomas relacionados con el mareo por movimiento: visión alterada, desorientación, alteraciones en el equilibrio graves mareos, convulsiones, espasmos ópticos o musculares, movimientos involuntarios [5]. No obstante, debido a la mejora en las prestaciones de los dispositivos, cada vez es menos frecuente que esto ocurra.
- El peso del HMD y a la utilización de mandos pueden producir dolores en músculos, articulaciones, cuello, manos o piel [5].
- Posibilidad de transmisión de enfermedades contagiosas al compartir el equipo entre varias personas [5].
- El usuario se encuentra inmerso en un entorno virtual, por lo que su uso prolongado puede producir problemas psicológicos desde distorsión de la realidad, hasta disparar o agravar desórdenes emocionales.
- Peligro de golpearse con objetos situados en la misma habitación y causar daños físicos graves.
- El usuario no es consciente de lo que sucede en el mundo real, debido a una inmersión en un entorno virtual. Esto puede generar dificultades en el caso de incendios, derrumbamientos, robos...

#### 4.1.4. Ejemplos de aplicaciones

En esta sección, se muestran ejemplos de aplicaciones de la tecnología de realidad virtual en distintos ámbitos:

- **Ocio.** Se aprovecha la tecnología de realidad virtual para el entretenimiento. El uso más conocido es el de la creación de videojuegos. Generalmente, se desarrollan versiones en realidad virtual de videojuegos ya existentes, como es el caso de Fruit Ninja VR. Además de los videojuegos, existen aplicaciones en visualización de contenidos multimedia, como Netflix VR.
- **Uso terapéutico.** Se crean entornos virtuales para el tratamiento de miedos y fobias [10, 13, 14], los cuales se basan en la exposición al elemento que produce el miedo en cuestión, así como para la superación de adicciones [15]. Otro ejemplo es el tratamiento de trastornos psicológicos como, por ejemplo, en casos de autismo severo [3].
- **Entrenamiento.** Se realizan simulaciones en realidad virtual como soporte al entrenamiento, con el objetivo de eliminar los riesgos que produce la práctica real de estas actividades. Existen aplicaciones en cirugía [4, 16, 17] o en pilotaje de aviones [12].
- **Fisioterapia.** Utilidad en rehabilitación de dolencias físicas, generando estímulos que fomenten una rápida recuperación [18].
- **Educación.** Se utilizan entornos virtuales para conseguir un aprendizaje interactivo, con la posibilidad de realizarse en cualquier lugar y en cualquier momento [2].
- **Turismo.** Se ofrecen nuevas experiencias como visitas virtuales a réplicas de museos [19, 20] o monumentos históricos [11].
- **Planificación colaborativa.** La realidad virtual permite la planificación colaborativa sin la necesidad de que los miembros del equipo de trabajo se encuentren en las mismas situaciones geográficas, con el fin de reducir los costes en movilidad [21, 22]. En este caso, se trata de aplicaciones similares a la propuesta realizada.

## 4.2. Motores de juego

En este apartado, se realiza una comparativa entre los motores de juego compatibles con realidad virtual. En este caso, se consideran las más conocidas en la actualidad, ya que son las que poseen una mayor comunidad, aspecto especialmente útil cuando es la primera vez que se utiliza un motor de juego para desarrollar una aplicación. Estos motores de juego son Unreal Engine 4 de Epic Games y Unity3D 5 de Unity Technologies.

Se trata de dos motores de juego consolidados en el desarrollo de aplicaciones, especialmente en el ámbito de los videojuegos. Por ello, la elección de cualquiera de ellos no afectará significativamente en el resultado obtenido. No obstante, es necesario analizar sus características principales para elegir el más adecuado para el desarrollo del proyecto. A continuación, se muestra una tabla comparativa entre ambos motores de juego.

TABLA 4.1: COMPARACIÓN ENTRE MOTORES DE JUEGO

	Unreal Engine	Unity3D
<b>VR soportado</b>	Sí	Sí
<b>Versión gratuita</b>	Sí	Sí
<b>Programación</b>	Orientada a diseñadores, no es necesario generar código	Orientada a programadores
<b>Lenguaje de programación</b>	C++	C#, Javascript
<b>Aprendizaje</b>	Mayor complejidad, mejor opción para personas experimentadas	Mayor sencillez, mejor opción para principiantes
<b>Documentación</b>	Manuales y tutoriales	Manuales y tutoriales
<b>Comunidad</b>	Activa, con foro propio	Activa, con foro propio
<b>Contenido de terceros</b>	Escaso	Amplio catálogo
<b>Gráficos</b>	Última generación	Limitados, pero utiliza menos recursos
<b>Multiplataforma</b>	Sí, pero principalmente PC y consolas	Sí
<b>Tipos de aplicaciones</b>	Aplicaciones basadas en FPS	Todo tipo de aplicaciones

En primer lugar, cabe destacar que las dos alternativas son válidas para el desarrollo del proyecto. Después de analizar las características principales de cada alternativa, se ha decidido utilizar Unity3D como motor de juego. La principal razón por la que se ha tomado esta decisión es por su sencillez, además de la gran comunidad de usuarios que posee. En el caso de Unreal Engine, aunque se trate de un motor de juego más completo, su aprendizaje es más complicado, ya que sus conceptos son más complejos y menos intuitivos. Por esta razón, se dice que Unity3D es la mejor opción para principiantes mientras que, con Unreal Engine, es necesario dedicarle más tiempo al aprendizaje.

Es cierto que Unreal Engine, además de un desarrollo basado en scripting, ofrece un desarrollo basado en blueprints, mediante el cual no es necesario ningún tipo de programación. Esta forma de desarrollar utiliza conceptos más cercanos a los diseñadores, dado que se basa en la utilización de nodos. Las competencias del Grado realizado son más cercanas a la programación y generación de código, por lo que no se considera una ventaja a considerar.

Otra característica que posiciona a Unity3D por encima de Unreal Engine de cara a este proyecto es el contenido de terceros. Unity posee una gran cantidad de librerías gratuitas que facilitan el desarrollo de la aplicación, frente a la alternativa de Epic Games, que requiere en la mayoría de los casos un desarrollo desde cero.

Por último, es necesario destacar la compatibilidad multiplataforma que ofrece Unity3D, destacando sobre su competidor en cuanto a dispositivos móviles. Este aspecto puede ser positivo de cara a futura compatibilidad del proyecto con otras plataformas.

### **4.3. Dispositivos**

A continuación, se detallarán los dispositivos de realidad virtual que se pueden utilizar para el desarrollo del proyecto. Debido a la inmersión del usuario al utilizar esta tecnología, los únicos dispositivos de realidad virtual comercializables son las gafas de realidad virtual o HMD. Se analizarán aquellos que sean compatibles con el motor de juego elegido.

#### **4.3.1. Carcasas o cascos sin pantalla**

Este tipo de HMD contiene un soporte en el que se puede colocar un smartphone, que servirá tanto de pantalla, como de motor de procesamiento. Se trata de la alternativa más económica y, por tanto, la que predomina en los hogares. Ejemplos de cascos sin pantalla son Cardboard y Daydream de Google, o Gear VR de Samsung.

La resolución de estos dispositivos es variable, dado que depende del smartphone que se utilice como pantalla. Además, no proporcionan mandos que contribuyan a formas de interacción que faciliten la inmersión del usuario.

Dadas las limitaciones de estos cascos, y debido a que el proyecto está ideado para un entorno profesional, en el que se suelen utilizar ordenadores de propósito general, además de ser los HMDs con mayor probabilidad de provocar mareos, esta opción queda descartada.

#### 4.3.2. Cascos con pantalla incorporada

En este caso, el motor de procesamiento será un ordenador o una consola. Teniendo en cuenta que el proyecto está enfocado a un entorno profesional, además de las limitaciones que ofrece una consola en cuanto a resolución, gráficos y personalización, se descartan aquellos HMD específicos para consolas como Sony PlayStation VR. Por ello, se analizarán aquellas gafas de realidad virtual que tengan soporte para PC.

A continuación, se muestra una tabla comparativa de las especificaciones de los cascos de realidad virtual con pantalla incorporada más relevantes.

TABLA 4.2: COMPARACIÓN ENTRE CASCOS DE REALIDAD VIRTUAL

	Oculus Rift	HTC Vive	HTC Vive Pro (abril 2018)	PIMAX 4k
<b>Resolución</b>	2160x1200	2160x1200	2880x1600	3840x2160
<b>Ratio de refresco</b>	90 Hz	90 Hz	90 Hz	60 Hz
<b>Tipo de pantalla</b>	PenTile OLED	PenTile OLED	AMOLED	LCD
<b>Campo de visión</b>	110°	110°	110°	110°
<b>Soporte para room scale</b>	Requiere un tercer sensor	Sí	Sí	Sí
<b>Rastreo de posición</b>	Sí	Sí	Sí	Requiere dispositivos externos
<b>Peso</b>	470 gramos	470 gramos	< 470 gramos, mejor distribución	490 gramos
<b>Auriculares integrados</b>	Sí	Opcional	Sí	Si
<b>Micrófono incorporado</b>	Sí	Sí	Sí	Sí
<b>Cámara externa</b>	No	Sí	Sí (dos)	No
<b>Conexión</b>	Cable, inalámbrica (adaptador)	Cable, inalámbrica (adaptador)	Cable, inalámbrica (adaptador)	Cable
<b>Controles oficiales</b>	Xbox One Controller, Oculus remote y Oculus Touch	Vive controller	Vive controller	Vive controller, Pimax controller
<b>Compatibilidad con tiendas</b>	SteamVR, Oculus Store	SteamVR, HTC Viveport	SteamVR, HTC Viveport	SteamVR
<b>Precio actual (€)</b>	449	599	879 + accesorios	299

En primer lugar, cabe destacar que ya se dispone del casco de realidad virtual de Oculus, en concreto el modelo Oculus Rift CV1. Por ello, únicamente se optará por otro dispositivo en el caso de que la diferencia sea significativa de cara a la realización del proyecto.

Aunque los cuatro cascos de realidad virtual analizados son similares en cuanto a especificaciones, algunos poseen características que les hacen destacar frente al resto. En el caso de PIMAX 4k, su resolución de pantalla es la más alta, ofreciendo una mejor calidad de visión, además de tratarse de la alternativa más económica. No obstante, su ratio de refresco de 60 Hz provoca una mayor facilidad de producir mareo o *cybersickness*. Teniendo en cuenta que, en ocasiones, en el desarrollo se le da un uso constante al casco de realidad virtual, se trata de una desventaja clara respecto al resto de alternativas.

En cuanto a HTC Vive y Oculus Rift, se trata de dos HMD con características prácticamente idénticas. Cuando HTC Vive se introdujo en el mercado, era superior a la versión de Oculus, debido al *room scale tracking* o a la posibilidad de conexión inalámbrica del dispositivo. Actualmente, Oculus ha solventado estas deficiencias y equilibra la elección entre una alternativa u otra.

En el caso de HTC Vive Pro, se trata de una versión mejorada de HTC Vive. En cuanto a especificaciones, destaca frente a la versión anterior en la resolución de pantalla, además de mejoras en ergonomía y un mayor equilibrio del peso del dispositivo. Esto último ayuda a prevenir el dolor de cuello que puede provocar el uso frecuente de estos dispositivos. El inconveniente principal de esta opción es su precio, además de que comenzó a estar disponible a partir de abril de 2018, a un mes de la finalización del proyecto. No obstante, se menciona en este apartado porque sería la opción elegida en un futuro en caso de una disminución de su precio.

Después de analizar todas las alternativas, se considera que el HMD Oculus Rift CV1 es válido para la realización del proyecto y que ninguna alternativa ofrece ventajas significativas. Por tanto, este será el dispositivo utilizado.

#### **4.4. Librerías**

En este apartado, se justifica la elección de los *assets* o librerías principales para llevar a cabo el desarrollo del proyecto.

##### **4.4.1. Realidad virtual**

En primer lugar, se analizaron las librerías que proporcionaran facilidades a la hora de desarrollar un proyecto en realidad virtual. Sin duda, la librería más completa en este aspecto es Virtual Reality Toolkit (VRTK), la cual es gratuita y compatible con el SDK de Oculus. Esta librería es útil para las interacciones con el mundo virtual, como el uso de rayos o el mostrado de unas manos virtuales en la posición donde se encuentran los controles, entre otros.

#### 4.4.2. Interacción con mapas

En segundo lugar, era necesaria una librería para la creación e interacción con mapas, independientemente de la inmersión del usuario. En este caso, Online Maps de Infinity Code, ha sido la opción elegida. Este software de pago ya se poseía con anterioridad, y se utilizó para experimentar con la interacción y visualización de mapas. Se trata de una opción que cumple sus funciones, y no se considera necesaria la adquisición de otra alternativa, ya que Online Maps es válida para el desarrollo del proyecto.

#### 4.4.3. Colaboración

Por último, debe realizarse un análisis del software necesario para la implementación de la parte colaborativa del proyecto (*networking*). Se consideran cuatro alternativas:

- **Unity Networking (UNet).** Se trata de la solución nativa de Unity, pensada para aplicaciones muy básicas con un número limitado de usuarios y, por tanto, que permite poca escalabilidad. Se considera que el resto de las alternativas son mejores.
- **Forge Networking.** Esta alternativa goza de una comunidad activa a través de Discord, y una documentación completa, acompañada de tutoriales básicos. No obstante, el servidor debe ejecutarse en una máquina propia.
- **DarkRift Networking.** En este caso que sí ofrece servidores en la nube de forma gratuita sin límite de usuarios, que ofrece sus servidores de forma gratuita para el despliegue de aplicaciones. Es rápido y ligero, con una buena documentación, pero con una comunidad escasa. Este hecho implica no tener el apoyo de otros desarrolladores en caso de que surja algún problema.
- **Photon Unity Networking (PUN).** Es la solución más utilizada, con una comunidad extensa, que tiene cerca de 300.000 desarrolladores. Se caracteriza por su escalabilidad y rapidez, además de incluir chat de texto y de voz. Al igual que DarkRift Networking, ofrece sus servidores en la nube para el despliegue, con la diferencia de que la versión gratuita se limita a 20 usuarios simultáneos (CCU). En caso de necesitar un mayor número de usuarios, será necesario comprar un plan.

Dadas las principales características de cada alternativa de colaboración, se ha decidido utilizar PUN, debido a su servicio gratuito en la nube y a la inclusión de chat de voz y texto, además de su gran comunidad.

Es cierto que los servidores gratuitos de Photon están limitados a 20 usuarios de forma simultánea, pero es suficiente para el desarrollo de un prototipo de la aplicación. Darkrift Networking también es una buena opción, pero, dado que es la primera vez que se trabaja en Unity, es más importante la presencia de una comunidad que realice tutoriales y pueda solucionar dudas relativas a la implementación. No obstante, es posible que se utilice esta alternativa en proyectos futuros, una vez se hayan adquirido conocimientos suficientes acerca de la creación de entornos colaborativos.

A continuación, se muestra una tabla comparativa de las alternativas descritas, destacando las características más relevantes de cara al desarrollo del proyecto:

TABLA 4.3: COMPARACIÓN ENTRE LIBRERÍAS DE COLABORACIÓN

	<b>UNet</b>	<b>Forge</b>	<b>DarkRift</b>	<b>PUN</b>
<b>Escalabilidad</b>	Baja	Media	Alta	Alta
<b>Documentación</b>	Mejorable	Detallada	Detallada	Detallada
<b>Comunidad</b>	Amplia y activa	Activa en Discord	Reducida	Amplia y activa
<b>Servidor en la nube</b>	No	No	Si	Sí
<b>Chat de texto y voz</b>	No	No	No	Sí



## 5. ANÁLISIS

La primera fase en el desarrollo de software es el análisis de requisitos. En esta fase, el objetivo es especificar cuál va a ser el funcionamiento del sistema a desarrollar, es decir, una especificación precisa de sus funcionalidades y restricciones.

En primer lugar, se realiza la especificación de requisitos de usuario, en el que se define de una forma general el alcance del sistema, así como un análisis del entorno en el que se utilizará la aplicación. A partir de esta especificación, se generarán los casos de uso que determinarán los diferentes escenarios que se pueden dar en su manejo.

A continuación, a partir de los apartados anteriores, se elaborará la especificación de requisitos de software, los cuales profundizarán en aspectos técnicos. Estos requisitos serán útiles de cara a la definición de la arquitectura del sistema.

Por último, se realiza un análisis de clases, cuya utilidad principal es conocer la información con la que se va a trabajar en la fase de implementación.

### 5.1. Descripción general de la aplicación

Se trata de una aplicación colaborativa de visualización e interacción con mapas. En ella, los usuarios se conectan a una misma sala, en la cual se situará un mapa. El mapa se dispone frente al usuario como un plano vertical que muestra la totalidad o una porción de su contenido, y que abarca todo el campo de visión del usuario. Cada usuario podrá elegir entre un entorno de realidad virtual con el HMD Oculus Rift o a través de un monitor corriente y una interacción con un teclado y un ratón. Los usuarios podrán trabajar juntamente con el mapa, independientemente del entorno elegido.

Los usuarios tendrán la capacidad de desplazar el contenido del mapa, así como cambiar el nivel de *zoom* de este. La posición de los usuarios es estática, de tal forma que, tanto para el desplazamiento como para cambiar el nivel de *zoom*, se modifica el contenido que presenta el plano que muestra el mapa. Asimismo, cada usuario navega de forma independiente por el mapa, de forma que cada uno de ellos observa el fragmento del mapa que desea.

La colaboración entre los usuarios se produce mediante el uso de marcadores. En el entorno de realidad virtual, se señala una posición en el mapa con el centro de la mirada o un rayo procedente del mando derecho, mientras que se utiliza el cursor del ratón en un entorno sin realidad virtual. Con esto, los usuarios podrán añadir, eliminar y destacar marcadores en el mapa. Destacar un marcador implica que este llame la atención visualmente respecto al resto y, si un marcador destacado se encuentra fuera del plano vertical, se mostrará una flecha que lo señale. Para distinguir las acciones de los usuarios, cada uno estará asociado a un color diferente, y este será el color de los marcadores que añada y el que indique qué usuario ha destacado un marcador. El uso de marcadores está asociado a señalar lugares del mapa, es decir, la aplicación no calcula rutas entre los marcadores ni muestra información relativa al lugar marcado. Adicionalmente, la aplicación permite la comunicación de los usuarios de la sala a través de un chat de voz, y muestra una lista de los usuarios que se encuentran en esta.

## **5.2. Requisitos de usuario**

A lo largo de este apartado, se definen las capacidades y restricciones generales del sistema, además de realizar un análisis del entorno de aplicación del producto software. En este análisis, se explicarán las características principales de los usuarios potenciales de la aplicación, así como aspectos relacionados con el entorno operativo y de seguridad y las suposiciones y dependencias encontradas de las que se parte a la hora de realizar el análisis del sistema.

### **5.2.1. Capacidades generales**

Entre las capacidades generales del sistema, se encuentran:

- Visualización e interacción con mapas en un entorno virtual a través del uso de marcadores. El sistema permitirá a los usuarios la posibilidad de añadir, eliminar y destacar marcadores.
- El entorno virtual es colaborativo, es decir, todos los usuarios observarán los marcadores que se hayan creado en el mapa. Estos marcadores se distinguirán por colores en función del usuario que los haya creado.
- Disponibilidad de un chat de voz que permite a los usuarios comunicarse entre ellos.

### **5.2.2. Restricciones generales**

Las restricciones generales de la aplicación a desarrollar son las siguientes:

- La utilización de la aplicación es principalmente a través de HMD de realidad virtual, pero se permite la interacción sin la utilización de estos dispositivos y, por tanto, sin la inmersión en un entorno virtual.
- Por cuestiones relativas a la tecnología utilizada, es imposible la realización tareas externas mientras el usuario se encuentre inmerso en el mundo virtual.
- La resolución de pantalla de los dispositivos de realidad virtual limita la nitidez con la que se observan los diferentes elementos del entorno virtual.
- La visualización del entorno virtual sólo será posible con la utilización de un HMD compatible de la marca Oculus, y a través de Sistema Operativo Windows.
- El desarrollo de la aplicación se realizará utilizando la plataforma de desarrollo y motor de juego Unity3D.

### **5.2.3. Características del usuario**

La aplicación está dirigida a personas que estén especializadas en tareas de visualización e interacción con mapas de forma colaborativa. Por ello, debe tener los conocimientos suficientes para interpretar el mapa mostrado por la aplicación.

En caso de que el usuario no esté familiarizado con la interacción en entornos virtuales, se deben proporcionar los mecanismos suficientes para aprender a utilizar los dispositivos de realidad virtual.

#### **5.2.4. Entorno operativo**

Para el funcionamiento del sistema, es necesario la ejecución de la aplicación en un ordenador características equivalentes o superiores:

- Conexión a internet. Es imprescindible para la comunicación entre los usuarios que utilicen la aplicación.
- Tarjeta gráfica NVIDIA GTX 1060 o AMD Radeon RX 480.
- CPU Intel i5-4590 o AMD Ryzen 5 1500X.
- Puerto HDMI 1.3 o adaptador compatible.
- 3 puertos USB 3.0 y 1 puerto USB 2.0.
- Sistema Operativo Windows 7 SP1 de 64 bits.

Las características discretas son los requisitos recomendados para la utilización del HMD Oculus Rift, excepto la necesidad de proporcionar conexión a internet. Esta información está disponible en la página web de Oculus [23].

#### **5.2.5. Entorno de seguridad**

Dado que no se considera el uso de credenciales ni el almacenamiento de datos personales, la aplicación no tiene ninguna responsabilidad en este aspecto.

#### **5.2.6. Suposiciones y dependencias**

A continuación, se describen las suposiciones y dependencias que se asumen a la hora de realizar el análisis del sistema:

- La aplicación se ejecutará en un ordenador que posea las características definidas en el entorno operativo.
- El sistema descrito no contempla el uso de credenciales para el acceso a la aplicación.
- La guía de uso de la aplicación proporcionada no incluye información acerca del uso de los dispositivos de realidad virtual.
- Se debe disponer de un dispositivo de realidad virtual compatible con Oculus, así como los accesorios necesarios para su uso, incluyendo sensores y los controles Oculus Touch.
- La aplicación se ejecutará en un entorno cerrado y sin obstáculos que puedan provocar posibles accidentes.
- El usuario poseerá un micrófono con el que podrá comunicarse vía voz con el resto de los usuarios.

### 5.2.7. Requisitos específicos

En este apartado, se detalla la especificación de un conjunto finito de requisitos a partir de la información descrita en los apartados anteriores. Se realiza una clasificación de los requisitos para permitir una identificación más sencilla:

- **Requisitos de capacidades.** Definen características o capacidades que debe poseer el sistema para ajustarse a las necesidades del usuario. Con ello, se consigue resolver un problema o lograr un objetivo. Estos requisitos se dividen a su vez en categorías, y cada una de ellas representa una temática diferente.
  - **Interacción con mapas.** Se trata de características relacionadas con la navegación a través del mapa.
  - **Uso de marcadores.** En este caso, se engloban aquellos requisitos relacionados con el manejo de los marcadores dispuestos en el mapa.
  - **Colaboración.** Esta categoría comprende las capacidades asociadas a la presencia de varios usuarios de forma simultánea en la aplicación.
  - **Otros requisitos.** Requisitos que no pertenezcan a ninguna de las categorías anteriores.
- **Requisitos de restricciones.** Son las limitaciones que condicionan cómo se debe resolver el problema o cómo se debe alcanzar un objetivo en concreto. Estas limitaciones pueden estar asociadas a la implementación, al rendimiento o a la disponibilidad del sistema, entre otros.

Asimismo, cada uno de los requisitos de usuario poseerá un conjunto de propiedades, a través de las cuales se llevará a cabo su especificación:

- **Identificador.** Cada requisito posee un identificador único que lo diferencia del resto de forma unívoca.  
El identificador poseerá la nomenclatura de tipo RU-TXX, la cual se descompone en los siguientes elementos:
  - RU: indica que se trata de un requisito de usuario.
  - T: indica el tipo de requisito. Puede tomar el valor C en el caso de ser un requisito de capacidad y el valor R si es un requisito de restricción.
  - XX: es el número que identifica al requisito dentro de cada tipo de requisito. Se representa como un número arábigo que comienza por el número uno (01) y se va incrementando de unidad en unidad.
- **Nombre.** Resumen claro y conciso del contenido del requisito.
- **Prioridad.** Indica preferencia temporal respecto a la realización de un requisito. Primero se tendrán en cuenta los requisitos con mayor prioridad. Podrá tomar tres valores: alta, media o baja.
- **Necesidad.** Indica si un requisito es fundamental en el desarrollo del producto (esencial) o puede llegar a omitirse (deseable).

- **Verificabilidad.** Es la facilidad con la que se puede comprobar que un requisito ha sido incorporado de forma correcta al sistema. Su verificabilidad será alta cuando sea fácilmente comprobable, media cuando su dificultad es media y baja cuando es difícil de comprobar. Se tratará de crear requisitos que sean altamente verificables.
- **Descripción.** Explicación detallada del contenido del requisito.

Todas estas propiedades se detallan en una tabla como la que se muestra a continuación:

TABLA 5.1: PLANTILLA DE REQUISITOS DE USUARIO

RU-C01	
Nombre	
Prioridad	
Necesidad	
Verificabilidad	
Descripción	

### 5.2.7.1. Requisitos de capacidad

A continuación, se incluye la especificación de requisitos de capacidad del conjunto de requisitos específicos de usuario, divididos en las categorías definidas en el apartado anterior.

#### a) Requisitos de interacción con mapas

TABLA 5.2: REQUISITO DE USUARIO RU-C01

RU-C01	
<b>Nombre</b>	Visualización de mapas
<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial
<b>Verificabilidad</b>	Alta
<b>Descripción</b>	El usuario tendrá la capacidad de observar y navegar a través de un mapa en dos dimensiones.

TABLA 5.3: REQUISITO DE USUARIO RU-C02

RU-C02	
<b>Nombre</b>	Ampliación del mapa
<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial
<b>Verificabilidad</b>	Alta
<b>Descripción</b>	El usuario podrá ampliar el mapa, de tal forma que se observe menor contenido, pero con un mayor grado de detalle.

TABLA 5.4: REQUISITO DE USUARIO RU-C03

<b>RU-C03</b>	
<b>Nombre</b>	Disminución de ampliación del mapa
<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial
<b>Verificabilidad</b>	Alta
<b>Descripción</b>	El usuario podrá alejar el mapa, de tal forma que se observe más contenido, pero con un menor grado de detalle.

TABLA 5.5: REQUISITO DE USUARIO RU-C04

<b>RU-C04</b>	
<b>Nombre</b>	Desplazamiento del mapa
<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial
<b>Verificabilidad</b>	Alta
<b>Descripción</b>	El usuario tendrá la capacidad de desplazar el contenido del mapa hacia arriba, abajo, derecha, izquierda o hacia una combinación de estas.

## b) Requisitos de uso de marcadores

TABLA 5.6: REQUISITO DE USUARIO RU-C05

<b>RU-C05</b>	
<b>Nombre</b>	Adición de marcadores
<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial
<b>Verificabilidad</b>	Alta
<b>Descripción</b>	El usuario podrá añadir marcadores, que se situarán en el mapa y señalarán un punto en concreto.

TABLA 5.7: REQUISITO DE USUARIO RU-C06

<b>RU-C06</b>	
<b>Nombre</b>	Eliminación de marcadores propios
<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial
<b>Verificabilidad</b>	Alta
<b>Descripción</b>	El usuario tendrá la capacidad de eliminar de forma individual aquellos marcadores que haya creado anteriormente.

TABLA 5.8: REQUISITO DE USUARIO RU-C07

<b>RU-C07</b>	
<b>Nombre</b>	Destacar marcadores
<b>Prioridad</b>	Media
<b>Necesidad</b>	Esencial
<b>Verificabilidad</b>	Alta
<b>Descripción</b>	El usuario podrá destacar de forma individual cualquier marcador situado en el mapa. Un marcador destacado o resaltado predominará de forma visual sobre aquellos que no lo estén.

TABLA 5.9: REQUISITO DE USUARIO RU-C08

<b>RU-C08</b>	
<b>Nombre</b>	Eliminación de destaque de marcadores
<b>Prioridad</b>	Media
<b>Necesidad</b>	Esencial
<b>Verificabilidad</b>	Alta
<b>Descripción</b>	El usuario podrá revertir el estado de un marcador destacado anteriormente por él mismo, de tal forma que se muestre visualmente tal y como era antes de destacarse.



TABLA 5.10: REQUISITO DE USUARIO RU-C09

<b>RU-C09</b>	
<b>Nombre</b>	Colores de los marcadores
<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial
<b>Verificabilidad</b>	Alta
<b>Descripción</b>	El color del marcador estará determinado por el usuario que lo haya creado. De esta forma, los marcadores creados por distintos usuarios serán de distinto color.

TABLA 5.11: REQUISITO DE USUARIO RU-C10

<b>RU-C10</b>	
<b>Nombre</b>	Eliminación de todos los marcadores propios
<b>Prioridad</b>	Baja
<b>Necesidad</b>	Deseable
<b>Verificabilidad</b>	Alta
<b>Descripción</b>	El usuario podrá eliminar con un comando todos los marcadores que haya creado anteriormente.

### c) Requisitos de colaboración

TABLA 5.12: REQUISITO DE USUARIO RU-C11

<b>RU-C11</b>	
<b>Nombre</b>	Visualización de marcadores de otros usuarios
<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial
<b>Verificabilidad</b>	Alta
<b>Descripción</b>	Cuando un usuario añada un marcador, el resto de los usuarios observarán el nuevo marcador en el mapa.

TABLA 5.13: REQUISITO DE USUARIO RU-C12

<b>RU-C12</b>	
<b>Nombre</b>	Eliminación de marcadores de otros usuarios
<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial
<b>Verificabilidad</b>	Alta
<b>Descripción</b>	Cuando un usuario elimine un marcador, este desaparecerá en el mapa del resto de usuarios.

TABLA 5.14: REQUISITO DE USUARIO RU-C13

<b>RU-C13</b>	
<b>Nombre</b>	Destaque de marcadores por otros usuarios
<b>Prioridad</b>	Media
<b>Necesidad</b>	Esencial
<b>Verificabilidad</b>	Alta
<b>Descripción</b>	Cuando un usuario destaque un marcador, el resto de los usuarios percibirán visualmente el marcador resaltado.

TABLA 5.15: REQUISITO DE USUARIO RU-C14

<b>RU-C14</b>	
<b>Nombre</b>	Número y posición de los marcadores
<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial
<b>Verificabilidad</b>	Alta
<b>Descripción</b>	Todos los usuarios observarán en sus respectivos mapas el mismo número de marcadores. Además, todos ellos señalarán los mismos lugares en el mapa.

TABLA 5.16: REQUISITO DE USUARIO RU-C15

<b>RU-C15</b>	
<b>Nombre</b>	Chat de voz
<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial
<b>Verificabilidad</b>	Alta
<b>Descripción</b>	El usuario tendrá la capacidad de comunicarse con el resto de los usuarios a través de la utilización de un chat de voz.

d) **Otros requisitos.**

TABLA 5.17: REQUISITO DE USUARIO RU-C16

<b>RU-C16</b>	
<b>Nombre</b>	Elección de entorno virtual
<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial
<b>Verificabilidad</b>	Alta
<b>Descripción</b>	El usuario podrá elegir entre la inmersión en un entorno de realidad virtual o un entorno real (de no realidad virtual).

TABLA 5.18: REQUISITO DE USUARIO RU-C17

<b>RU-C17</b>	
<b>Nombre</b>	Agrupación de usuarios en salas
<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial
<b>Verificabilidad</b>	Alta
<b>Descripción</b>	Los usuarios se agrupan en salas. Una sala es el entorno en el cual un grupo de usuarios se comunican y colaboran interaccionando con el mismo mapa.

### 5.2.7.2. Requisitos de restricción

En este apartado, se detalla la especificación de requisitos de restricción del conjunto de requisitos específicos de usuario.

TABLA 5.19: REQUISITO DE USUARIO RU-R01

<b>RU-R01</b>	
<b>Nombre</b>	Plataforma de desarrollo
<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial
<b>Verificabilidad</b>	Alta
<b>Descripción</b>	El desarrollo de la aplicación se realizará utilizando el motor de juego Unity3D.

TABLA 5.20: REQUISITO DE USUARIO RU-R02

<b>RU-R02</b>	
<b>Nombre</b>	Aplicación de realidad virtual
<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial
<b>Verificabilidad</b>	Alta
<b>Descripción</b>	La aplicación debe ser compatible con la tecnología de realidad virtual, es decir, debe poder observarse el entorno generado utilizando un HMD de realidad virtual.

TABLA 5.21: REQUISITO DE USUARIO RU-R03

<b>RU-R03</b>	
<b>Nombre</b>	Compatibilidad con Oculus
<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial
<b>Verificabilidad</b>	Alta
<b>Descripción</b>	La aplicación debe ser compatible con el HMD Oculus Rift CV1.

TABLA 5.22: REQUISITO DE USUARIO RU-R04

<b>RU-R04</b>	
<b>Nombre</b>	Extensión a entornos de no realidad virtual
<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial
<b>Verificabilidad</b>	Alta
<b>Descripción</b>	Debe ser posible la ejecución de la aplicación en entornos que no sean de realidad virtual, es decir, debe poder visualizarse a través de monitores o pantallas.

TABLA 5.23: REQUISITO DE USUARIO RU-R05

<b>RU-R05</b>	
<b>Nombre</b>	Controles de realidad virtual
<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial
<b>Verificabilidad</b>	Alta
<b>Descripción</b>	La interacción entre el usuario y la aplicación en el entorno de realidad virtual debe realizarse utilizando los controles Oculus Touch.

TABLA 5.24: REQUISITO DE USUARIO RU-R06

<b>RU-R06</b>	
<b>Nombre</b>	Controles de no realidad virtual
<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial
<b>Verificabilidad</b>	Alta
<b>Descripción</b>	La interacción entre el usuario y la aplicación en el entorno que no es de realidad virtual debe realizarse utilizando ratón y teclado.

TABLA 5.25: REQUISITO DE USUARIO RU-R07

<b>RU-R07</b>	
<b>Nombre</b>	Colaboración entre diferentes entornos virtuales
<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial
<b>Verificabilidad</b>	Alta
<b>Descripción</b>	Debe existir colaboración entre usuarios en entornos de realidad virtual y no realidad virtual.

TABLA 5.26: REQUISITO DE USUARIO RU-R08

<b>RU-R08</b>	
<b>Nombre</b>	Desarrollo de realidad virtual
<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial
<b>Verificabilidad</b>	Alta
<b>Descripción</b>	El desarrollo de realidad virtual debe realizarse en un ordenador que posea las especificaciones recomendadas por Oculus o superiores.

TABLA 5.27: REQUISITO DE USUARIO RU-R09

<b>RU-R09</b>	
<b>Nombre</b>	Acceso a internet
<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial
<b>Verificabilidad</b>	Alta
<b>Descripción</b>	La aplicación debe tener acceso a internet para poder llevar a cabo la colaboración.

TABLA 5.28: REQUISITO DE USUARIO RU-R10

<b>RU-R10</b>	
<b>Nombre</b>	Visualización de la información
<b>Prioridad</b>	Media
<b>Necesidad</b>	Esencial
<b>Verificabilidad</b>	Media
<b>Descripción</b>	La información mostrada en la ejecución de la aplicación debe ser nítida y legible en caso de tratarse de texto.

TABLA 5.29: REQUISITO DE USUARIO RU-R11

<b>RU-R11</b>	
<b>Nombre</b>	Respuesta de la aplicación
<b>Prioridad</b>	Media
<b>Necesidad</b>	Esencial
<b>Verificabilidad</b>	Media
<b>Descripción</b>	La aplicación debe ofrecer una respuesta rápida ante las acciones realizadas por el usuario, de tal forma que el usuario no se frustre y/o dificulte la inmersión en un entorno virtual.

TABLA 5.30: REQUISITO DE USUARIO RU-R12

<b>RU-R12</b>	
<b>Nombre</b>	Intuitividad
<b>Prioridad</b>	Media
<b>Necesidad</b>	Esencial
<b>Verificabilidad</b>	Media
<b>Descripción</b>	La aplicación debe ser intuitiva, facilitando el aprendizaje de los controles. Además, los elementos gráficos mostrados deben expresar correctamente su significado.



### 5.3. Casos de uso

A partir de los requisitos de usuario que se han generado en el apartado anterior, se especificarán los casos de uso. Los casos de uso simularán todos los escenarios posibles que pueden darse al utilizar la aplicación. Del mismo modo, aportan información útil de cara a la especificación de requisitos de software.

A continuación, se muestra el diagrama de casos de uso generado a partir de los requisitos de usuario. Cabe destacar que las directivas de tipo *include* indican la necesidad de haber realizado otro caso de uso para que este no devuelva un error. Por ejemplo, para poder eliminar un marcador propio, es necesario haber añadido un marcador. Si no se ha añadido ningún marcador antes, el sistema siempre indicará el error.



Fig. 5.1. Diagrama de casos de uso

La especificación de cada caso de uso consta de una tabla con las siguientes propiedades:

- **Identificador.** Cada caso de uso posee un identificador único. La nomenclatura utilizada es del formato CU-XX, donde CU indica que se trata de un caso de uso, y XX representa un número entero de dos cifras que comienza por el número uno (01) y se va incrementando de unidad en unidad.
- **Nombre.** Descripción breve del caso de uso.
- **Flujo principal.** Se enumeran los pasos a seguir para llevar a cabo el caso de uso.
- **Flujos alternativos.** Secuencia de pasos alternativa en la que se obtiene el mismo resultado.
- **Precondiciones.** Condiciones que se deben cumplir para realizar la operación, entre las que se encuentran aquellos casos de uso de los que depende su ejecución (directivas *include* en el diagrama).
- **Postcondiciones.** Son los efectos de realizar la operación, y corresponde con la respuesta del sistema.
- **Excepciones.** Descripción textual de las condiciones de fallo que afectan al escenario y las respuestas del sistema ante esas situaciones.
- **Fuente.** Requisitos de usuario asociados al caso de uso.

A partir de estas propiedades, se crea la plantilla de la tabla que se va a utilizar para especificar cada caso de uso:

TABLA 5.31: PLANTILLA DE CASOS DE USO

CU-XX		
<b>Nombre</b>		
<b>Flujo principal</b>	<b>1</b>	Acción 1
	<b>2</b>	Acción 2
<b>Flujo alternativo 1</b>	<b>1</b>	Acción 1
	...	...
<b>Flujo alternativo 2</b>	...	...
<b>Precondiciones</b>		
<b>Postcondiciones</b>		
<b>Excepciones</b>		
<b>Fuente</b>		

A continuación, se procede a detallar la especificación de los casos de uso que se han mostrado en el diagrama:

TABLA 5.32: CASO DE USO CU-01

<b>CU-01</b>		
<b>Nombre</b>	<i>Zoom in</i>	
<b>Escenario principal</b>	<b>1</b>	Señalar posición en el mapa con la mirada ( <i>gaze</i> ).
	<b>2</b>	Utilizar Oculus Touch para ampliar el mapa.
<b>Flujo alternativo 1</b>	<b>1</b>	Utilizar Oculus Touch para activar <i>gaze</i> (desactivar rayo).
	<b>2</b>	Señalar posición en el mapa con la mirada ( <i>gaze</i> )
	<b>3</b>	Utilizar Oculus Touch para ampliar el mapa.
<b>Flujo alternativo 2</b>	<b>1</b>	Señalar posición en el mapa con el rayo.
	<b>2</b>	Utilizar Oculus Touch para ampliar el mapa.
<b>Flujo alternativo 3</b>	<b>1</b>	Utilizar Oculus Touch para activar rayo (desactivar <i>gaze</i> ).
	<b>2</b>	Señalar posición en el mapa con el rayo.
	<b>3</b>	Utilizar Oculus Touch para ampliar el mapa.
<b>Flujo alternativo 4</b>	<b>1</b>	Señalar posición en el mapa con el ratón.
	<b>2</b>	Utilizar ratón para ampliar el mapa.
<b>Precondiciones</b>	La aplicación se encuentra iniciada y se puede observar el mapa. En el caso de realidad virtual, el HMD debe estar colocado en la cabeza del usuario.	
<b>Postcondiciones</b>	El mapa se amplía con la posición señalada como centro del fragmento ampliado.	
<b>Excepciones</b>	El nivel de <i>zoom</i> antes de ejecutar el escenario es el máximo. El sistema informa al usuario del error de forma visual y/o auditiva.	
<b>Fuente</b>	RU-C01, RU-C02, RU-C16, RU-C17	

TABLA 5.33: CASO DE USO CU-02

<b>CU-02</b>		
<b>Nombre</b>	<i>Zoom out</i>	
<b>Flujo principal</b>	<b>1</b>	Señalar posición en el mapa con la mirada ( <i>gaze</i> ).
	<b>2</b>	Utilizar Oculus Touch para alejar el mapa.
<b>Flujo alternativo 1</b>	<b>1</b>	Utilizar Oculus Touch para activar <i>gaze</i> (desactivar rayo).
	<b>2</b>	Señalar posición en el mapa con la mirada ( <i>gaze</i> ).
	<b>3</b>	Utilizar Oculus Touch para alejar el mapa.
<b>Flujo alternativo 2</b>	<b>1</b>	Señalar posición en el mapa con el rayo.
	<b>2</b>	Utilizar Oculus Touch para alejar el mapa.
<b>Flujo alternativo 3</b>	<b>1</b>	Utilizar Oculus Touch para activar rayo (desactivar <i>gaze</i> ).
	<b>2</b>	Señalar posición en el mapa con el rayo.
	<b>3</b>	Utilizar Oculus Touch para alejar el mapa.
<b>Flujo alternativo 4</b>	<b>1</b>	Señalar posición en el mapa con el ratón.
	<b>2</b>	Utilizar ratón para alejar el mapa.
<b>Precondiciones</b>	La aplicación se encuentra iniciada y se puede observar el mapa. En el caso de realidad virtual, el HMD debe estar colocado en la cabeza del usuario.	
<b>Postcondiciones</b>	El mapa se aleja con la posición señalada como centro del fragmento alejado.	
<b>Excepciones</b>	El nivel de <i>zoom</i> antes de darse el escenario es el mínimo. El sistema informa al usuario del error de forma visual y/o auditiva.	
<b>Fuente</b>	RU-C01, RU-C03, RU-C16, RU-C17	

TABLA 5.34: CASO DE USO CU-03

<b>CU-03</b>		
<b>Nombre</b>	Desplazar el mapa	
<b>Flujo principal</b>	<b>1</b>	Utilizar Oculus Touch para desplazar el mapa hacia la dirección deseada (arriba, abajo, derecha, izquierda o una combinación de estas).
<b>Flujo alternativo</b>	<b>1</b>	Utilizar teclado para desplazar el mapa hacia la dirección deseada (arriba, abajo, derecha, izquierda o una combinación de estas).
<b>Precondiciones</b>	La aplicación se encuentra iniciada y se puede observar el mapa. En el caso de realidad virtual, el HMD debe estar colocado en la cabeza del usuario.	
<b>Postcondiciones</b>	El contenido del mapa se desplaza en la dirección indicada, ocultando información de la dirección elegida y mostrando nueva información en el sentido opuesto.	
<b>Excepciones</b>	Se ha alcanzado el límite superior o inferior del mapa. No se produce desplazamiento del mapa.	
<b>Fuente</b>	RU-C01, RU-C04, RU-C16, RU-C17	

TABLA 5.35: CASO DE USO CU-04

<b>CU-04</b>		
<b>Nombre</b>	Eliminar todos los marcadores propios	
<b>Flujo principal</b>	<b>1</b>	Utilizar Oculus Touch para eliminar todos los marcadores de los que el usuario es propietario
<b>Flujo alternativo</b>	<b>1</b>	Utilizar ratón para eliminar todos los marcadores de los que el usuario es propietario
<b>Precondiciones</b>	La aplicación se encuentra iniciada y se puede observar el mapa. En el caso de realidad virtual, el HMD debe estar colocado en la cabeza del usuario.	
<b>Postcondiciones</b>	Se eliminan todos los marcadores de los que el usuario es propietario	
<b>Excepciones</b>	Ninguna	
<b>Fuente</b>	RU-C01, RU-C10, RU-C12, RU-C14, RU-C16, RU-C17	

TABLA 5.36: CASO DE USO CU-05

CU-05		
<b>Nombre</b>	Añadir un marcador	
<b>Escenario principal</b>	<b>1</b>	Señalar posición en el mapa con la mirada ( <i>gaze</i> ).
	<b>2</b>	Utilizar Oculus Touch para añadir el marcador.
<b>Flujo alternativo 1</b>	<b>1</b>	Utilizar Oculus Touch para activar <i>gaze</i> (desactivar rayo).
	<b>2</b>	Señalar posición en el mapa con la mirada ( <i>gaze</i> ).
	<b>3</b>	Utilizar Oculus Touch para añadir el marcador.
<b>Flujo alternativo 2</b>	<b>1</b>	Señalar posición en el mapa con el rayo.
	<b>2</b>	Utilizar Oculus Touch para añadir el marcador.
<b>Flujo alternativo 3</b>	<b>1</b>	Utilizar Oculus Touch para activar rayo (desactivar <i>gaze</i> ).
	<b>2</b>	Señalar posición en el mapa con el rayo.
	<b>3</b>	Utilizar Oculus Touch para añadir el marcador.
<b>Flujo alternativo 4</b>	<b>1</b>	Señalar posición en el mapa con el ratón.
	<b>2</b>	Utilizar ratón para añadir el marcador.
<b>Precondiciones</b>	La aplicación se encuentra iniciada y se puede observar el mapa. En el caso de realidad virtual, el HMD debe estar colocado en la cabeza del usuario.	
<b>Postcondiciones</b>	El marcador se añade en la posición señalada.	
<b>Excepciones</b>	Ya existe un marcador en la posición indicada. El sistema informa al usuario del error de forma visual y/o auditiva.	
<b>Fuente</b>	RU-C01, RU-C05, RU-C09, RU-C11, RU-C14, RU-C16, RU-C17	

TABLA 5.37: CASO DE USO CU-06

<b>CU-06</b>		
<b>Nombre</b>	Eliminar los marcadores propios	
<b>Escenario principal</b>	<b>1</b>	Señalar marcador propio con la mirada ( <i>gaze</i> ).
	<b>2</b>	Utilizar Oculus Touch para eliminar el marcador.
<b>Flujo alternativo 1</b>	<b>1</b>	Utilizar Oculus Touch para activar <i>gaze</i> (desactivar rayo).
	<b>2</b>	Señalar marcador propio con la mirada ( <i>gaze</i> ).
	<b>3</b>	Utilizar Oculus Touch para eliminar el marcador.
<b>Flujo alternativo 2</b>	<b>1</b>	Señalar marcador propio con el rayo.
	<b>2</b>	Utilizar Oculus Touch para eliminar el marcador.
<b>Flujo alternativo 3</b>	<b>1</b>	Utilizar Oculus Touch para activar rayo (desactivar <i>gaze</i> ).
	<b>2</b>	Señalar marcador propio con el rayo.
	<b>3</b>	Utilizar Oculus Touch para eliminar el marcador.
<b>Flujo alternativo 4</b>	<b>1</b>	Señalar marcador propio con el ratón.
	<b>2</b>	Utilizar ratón para eliminar el marcador.
<b>Precondiciones</b>	La aplicación se encuentra iniciada y se puede observar el mapa. En el caso de realidad virtual, el HMD debe estar colocado en la cabeza del usuario. El usuario debe haber añadido al menos un marcador en el mapa.	
<b>Postcondiciones</b>	El marcador señalado es eliminado del mapa.	
<b>Excepciones</b>	Se señala una posición donde no existe ningún marcador o un marcador del que no se es propietario. El sistema informa al usuario del error de forma visual y/o auditiva.	
<b>Fuente</b>	RU-C01, RU-C06, RU-C12, RU-C14, RU-C16, RU-C17	

TABLA 5.38: CASO DE USO CU-07

CU-07		
Nombre	Destacar un marcador	
Escenario principal	1	Señalar marcador con la mirada ( <i>gaze</i> ).
	2	Utilizar Oculus Touch para destacar el marcador
Flujo alternativo 1	1	Utilizar Oculus Touch para activar <i>gaze</i> (desactivar rayo).
	2	Señalar marcador con la mirada ( <i>gaze</i> ).
	3	Utilizar Oculus Touch para destacar el marcador.
Flujo alternativo 2	1	Señalar marcador con el rayo.
	2	Utilizar Oculus Touch para destacar el marcador.
Flujo alternativo 3	1	Utilizar Oculus Touch para activar rayo (desactivar <i>gaze</i> ).
	2	Señalar marcador con el rayo.
	3	Utilizar Oculus Touch para destacar el marcador.
Flujo alternativo 4	1	Señalar marcador con el ratón.
	2	Utilizar ratón para destacar el marcador.
Precondiciones	La aplicación se encuentra iniciada y se puede observar el mapa. En el caso de realidad virtual, el HMD debe estar colocado en la cabeza del usuario. Debe existir al menos un marcador en el mapa.	
Postcondiciones	El marcador señalado es destacado, y este incluye el color asociado al usuario que lo ha destacado.	
Excepciones	Se señala una posición donde no existe ningún marcador. El sistema informa al usuario del error de forma visual y/o auditiva.	
Fuente	RU-C01, RU-C07, RU-C09, RU-C13, RU-C16, RU-C17	



TABLA 5.39: CASO DE USO CU-08

<b>CU-08</b>		
<b>Nombre</b>	Eliminar el destaque de un marcador	
<b>Escenario principal</b>	<b>1</b>	Señalar marcador destacado por el usuario con la mirada ( <i>gaze</i> ).
	<b>2</b>	Utilizar Oculus Touch para eliminar el destaque del marcador
<b>Flujo alternativo 1</b>	<b>1</b>	Utilizar Oculus Touch para activar <i>gaze</i> (desactivar rayo).
	<b>2</b>	Señalar marcador destacado por el usuario con la mirada ( <i>gaze</i> ).
	<b>3</b>	Utilizar Oculus Touch para eliminar el destaque del marcador.
<b>Flujo alternativo 2</b>	<b>1</b>	Señalar marcador destacado por el usuario con el rayo.
	<b>2</b>	Utilizar Oculus Touch para eliminar el destaque del marcador.
<b>Flujo alternativo 3</b>	<b>1</b>	Utilizar Oculus Touch para activar rayo (desactivar <i>gaze</i> ).
	<b>2</b>	Señalar marcador destacado por el usuario con el rayo.
	<b>3</b>	Utilizar Oculus Touch para eliminar el destaque del marcador.
<b>Flujo alternativo 4</b>	<b>1</b>	Señalar marcador destacado por el usuario con el ratón.
	<b>2</b>	Utilizar ratón para eliminar el destaque del marcador.
<b>Precondiciones</b>	La aplicación se encuentra iniciada y se puede observar el mapa. En el caso de realidad virtual, el HMD debe estar colocado en la cabeza del usuario. Debe existir al menos un marcador en el mapa.	
<b>Postcondiciones</b>	El marcador señalado es destacado, y este incluye el color asociado al usuario que lo ha destacado.	
<b>Excepciones</b>	Se señala una posición donde no existe ningún marcador. El sistema informa al usuario del error de forma visual y/o auditiva. En caso de que el marcador no estuviera destacado por el usuario, se obtiene el resultado del caso de uso CU-C07.	
<b>Fuente</b>	RU-C01, RU-C08, RU-C13, RU-C16, RU-C17	

TABLA 5.40: CASO DE USO CU-09

<b>CU-09</b>		
<b>Nombre</b>	Silenciar micrófono	
<b>Flujo principal</b>	<b>1</b>	Utilizar Oculus Touch para silenciar al micrófono.
<b>Flujo alternativo</b>	<b>1</b>	Utilizar teclado para silenciar el micrófono.
<b>Precondiciones</b>	La aplicación se encuentra iniciada y se puede observar el mapa. En el caso de realidad virtual, el HMD debe estar colocado en la cabeza del usuario. El micrófono debe estar activado.	
<b>Postcondiciones</b>	Se silencia el micrófono, de tal forma que el resto de los usuarios no pueden escucharle a través del chat de voz.	
<b>Excepciones</b>	Ninguna	
<b>Fuente</b>	RU-C15, RU-C16, RU-C17	

TABLA 5.41: CASO DE USO CU-10

<b>CU-10</b>		
<b>Nombre</b>	Activar el micrófono	
<b>Flujo principal</b>	<b>1</b>	Utilizar Oculus Touch para activar al micrófono.
<b>Flujo alternativo</b>	<b>1</b>	Utilizar teclado para activar el micrófono.
<b>Precondiciones</b>	La aplicación se encuentra iniciada y se puede observar el mapa. En el caso de realidad virtual, el HMD debe estar colocado en la cabeza del usuario. El micrófono debe estar silenciado.	
<b>Postcondiciones</b>	Se activa el micrófono, de tal forma que el resto de los usuarios pueden escucharle a través del chat de voz.	
<b>Excepciones</b>	Ninguna	
<b>Fuente</b>	RU-C15, RU-C16, RU-C17	

## 5.4. Requisitos de software

Una vez se han especificado los requisitos de usuario y los casos de uso, es posible profundizar en las capacidades y restricciones del sistema, dando lugar a los requisitos de software. Estos requisitos serán más detallados y contemplarán todas las posibilidades. Asimismo, se centrarán en cómo solucionar los problemas planteados.

Al igual que los requisitos específicos de usuario, se dividen en un conjunto de categorías que facilitan su identificación:

- **Requisitos funcionales (F).** Se listan las características y capacidades del software a desarrollar. Derivan de los casos de uso y de los requisitos de capacidad, de tal forma que se dividen en las mismas categorías.
  - **Interacción con mapas.** Se trata de características relacionadas con la navegación a través del mapa.
  - **Uso de marcadores.** En este caso, se engloban aquellos requisitos relacionados con el manejo de los marcadores dispuestos en el mapa.
  - **Colaboración.** Esta categoría comprende las capacidades asociadas a la presencia de varios usuarios de forma simultánea en la aplicación.
  - **Otros requisitos funcionales.** Requisitos que no pertenezcan a ninguna de las categorías anteriores.
- **Requisitos de rendimiento (RE).** Definen valores numéricos para variables de rendimiento.
- **Requisitos operacionales (OP).** Indican cómo el sistema va a realizar las tareas para las que ha sido construido, garantizando los niveles de servicio requeridos.
- **Requisitos de calidad (CA).** Especifican los atributos que debe poseer el software para ser adecuado para su propósito.
- **Requisitos de interfaz (I).** Se trata del hardware o software con el que el sistema o componentes del sistema deben interactuar o comunicarse.
- **Requisitos de usabilidad (U).** Aquellos requisitos relativos a la facilidad de uso que ofrece el sistema al usuario.

Asimismo, los requisitos especificados poseen un conjunto de propiedades que los definen en su totalidad. En este caso, son similares a las utilizadas para especificar los requisitos de usuario:

- **Identificador.** Cada requisito posee un identificador único que lo diferencia del resto de forma unívoca.  
El identificador poseerá la nomenclatura de tipo RS-TXX, la cual se descompone en los siguientes elementos:
  - RS: indica que se trata de un requisito de usuario.

- T: indica el tipo de requisito. Su valor depende de la categoría a la que pertenezca el requisito, y es la abreviatura utilizada al definir las diferentes categorías.
- XX: es el número que identifica al requisito dentro de cada tipo de requisito. Se representa como un número arábigo que comienza por el número uno (01) y se va incrementando de unidad en unidad.
- **Nombre.** Resumen claro y conciso del contenido del requisito.
- **Fuente.** Nueva propiedad respecto a los requisitos de usuario. Se trata del requisito de usuario que origina el requisito de software.
- **Prioridad.** Indica preferencia temporal respecto a la realización de un requisito. Primero se tendrán en cuenta los requisitos con mayor prioridad. Podrá tomar tres valores: alta, media o baja.
- **Necesidad.** Indica si un requisito es fundamental en el desarrollo del producto (esencial) o puede llegar a omitirse (deseable).
- **Verificabilidad.** Es la facilidad con la que se puede comprobar que un requisito ha sido incorporado de forma correcta al sistema. Su verificabilidad será alta cuando sea fácilmente comprobable, media cuando su dificultad es media y baja cuando es difícil de comprobar. Se tratará de crear requisitos que sean altamente verificables.
- **Descripción.** Explicación detallada del contenido del requisito.

Todas estas propiedades se detallan en una tabla como la que se muestra a continuación:

TABLA 5.42: PLANTILLA DE REQUISITOS DE SOFTWARE

RS-TXX			
Nombre			
Fuente		Prioridad	
Necesidad		Verificabilidad	
Descripción			

### 5.4.1. Requisitos funcionales

#### a) Requisitos de interacción con mapas

TABLA 5.43: REQUISITO DE SOFTWARE RS-F01

RS-F01			
<b>Nombre</b>	Visualización de mapas		
<b>Fuente</b>	RU-C01	<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	El usuario tendrá la capacidad de observar y navegar a través de un mapa en dos dimensiones.		

TABLA 5.44: REQUISITO DE SOFTWARE RS-F02

RS-F02			
<b>Nombre</b>	Ampliación del mapa ( <i>zoom in</i> )		
<b>Fuente</b>	RU-C02	<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	El usuario podrá ampliar el mapa de forma gradual, de tal forma que se observe menor contenido, pero con un mayor grado de detalle.		

TABLA 5.45: REQUISITO DE SOFTWARE RS-F03

RS-F03			
<b>Nombre</b>	Disminución de ampliación del mapa ( <i>zoom out</i> )		
<b>Fuente</b>	RU-C03	<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	El usuario podrá alejar el contenido del mapa, de tal forma que se observe más contenido alrededor de la zona donde se realiza <i>zoom out</i> . El resultado obtenido es un fragmento de mapa que abarca más lugares, pero con un menor grado de detalle.		

TABLA 5.46: REQUISITO DE SOFTWARE RS-F04

RS-F04			
<b>Nombre</b>	Niveles de <i>zoom</i>		
<b>Fuente</b>	RU-C02, RU-C03	<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	El grado de ampliación o disminución de la ampliación gradual se representará de forma numérica. Este número representará el nivel de <i>zoom</i> , y cuanto mayor sea su valor mayor grado de ampliación se dará en el mapa, y viceversa.		

TABLA 5.47: REQUISITO DE SOFTWARE RS-F05

RS-F05			
<b>Nombre</b>	Límite de ampliación del mapa		
<b>Fuente</b>	RU-C02	<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	Existe un límite superior de ampliación del mapa, en el cual se observará una porción específica de este con el mayor nivel de detalle posible.		

TABLA 5.48: REQUISITO DE SOFTWARE RS-F06

RS-F06			
<b>Nombre</b>	Límite de disminución de ampliación del mapa		
<b>Fuente</b>	RU-C03	<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	Existe un límite superior de disminución de ampliación del mapa, es decir, un límite inferior en la ampliación. En este límite, será posible observar el mapa en su completitud.		

TABLA 5.49: REQUISITO DE SOFTWARE RS-F07

RS-F07			
<b>Nombre</b>	Superación de los límites de ampliación		
<b>Fuente</b>	RU-C02, RU-C03	<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	Si se intenta sobrepasar el límite de ampliación, sea superior o inferior, se indicará al usuario la imposibilidad de realizar la acción.		

TABLA 5.50: REQUISITO DE SOFTWARE RS-F08

RS-F08			
<b>Nombre</b>	Desplazamiento del mapa		
<b>Fuente</b>	RU-C04	<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	El usuario tendrá la capacidad de desplazar el contenido del mapa hacia arriba, abajo, derecha, izquierda o hacia una combinación de estas. Cualquier desplazamiento implica ocultar parte de la información del mapa para añadir en el lado opuesto de la escena nueva información.		

TABLA 5.51: REQUISITO DE SOFTWARE RS-F09

RS-F09			
<b>Nombre</b>	Interacción individual con el mapa		
<b>Fuente</b>	RU-C01	<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	Cada usuario interactuará de forma individual con su mapa, de tal forma que dos usuarios pueden observar distintos puntos de este y con distinto nivel de <i>zoom</i> .		

## b) Requisitos de uso de marcadores

TABLA 5.52: REQUISITO DE SOFTWARE RS-F10

RS-F10			
<b>Nombre</b>	Adición de marcadores		
<b>Fuente</b>	RU-C05	<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	El usuario podrá añadir marcadores, que se situarán en el mapa y señalarán un punto en concreto.		

TABLA 5.53: REQUISITO DE SOFTWARE RS-F11

RS-F11			
<b>Nombre</b>	Identificador de usuario		
<b>Fuente</b>	RU-C05	<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	Cada usuario poseerá un identificador numérico que lo diferencia del resto de forma unívoca.		

TABLA 5.54: REQUISITO DE SOFTWARE RS-F12

RS-F12			
<b>Nombre</b>	Propietario de marcador		
<b>Fuente</b>	RU-C05	<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	Cada marcador tendrá asociado el identificador del usuario que lo ha creado.		



TABLA 5.55: REQUISITO DE SOFTWARE RS-F13

RS-F13			
<b>Nombre</b>	Posición del marcador (identificador)		
<b>Fuente</b>	RU-C05	<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	Cada marcador almacenará la posición en la que este es colocado. No existirán dos marcadores en la misma posición, por lo que la posición del marcador es su identificador.		

TABLA 5.56: REQUISITO DE SOFTWARE RS-F14

RS-F14			
<b>Nombre</b>	Eliminación de marcadores propios		
<b>Fuente</b>	RU-C06	<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	El usuario tendrá la capacidad de eliminar de forma individual aquellos marcadores que haya creado anteriormente, es decir, todos aquellos marcadores cuyo identificador del propietario coincida con el identificador del usuario.		

TABLA 5.57: REQUISITO DE SOFTWARE RS-F15

RS-F15			
<b>Nombre</b>	Imposibilidad de eliminación de marcadores no propios		
<b>Fuente</b>	RU-C06	<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	Si se intenta eliminar un marcador que no es propio (el identificador del usuario no coincide con el del propietario del marcador), se indicará al usuario la imposibilidad de realizar la acción.		

TABLA 5.58: REQUISITO DE SOFTWARE RS-F16

RS-F16			
<b>Nombre</b>	Marcador señalado		
<b>Fuente</b>	RU-C06	<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	Un usuario podrá señalar un marcador cuando le apunta utilizando algunos de los métodos de entrada disponibles. El marcador cambiará de color, indicando que se pueden realizar acciones sobre él.		

TABLA 5.59: REQUISITO DE SOFTWARE RS-F17

RS-F17			
<b>Nombre</b>	Señalamiento de posición o marcador en realidad virtual		
<b>Fuente</b>	RU-C06	<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	El usuario podrá alternar entre dos métodos de entrada, a través de los cuales puede señalar un marcador o una posición en el mapa en un entorno de realidad virtual: a través de un rayo que salga del usuario o utilizando el centro de la mirada.		

TABLA 5.60: REQUISITO DE SOFTWARE RS-F18

RS-F18			
<b>Nombre</b>	Destacar marcadores		
<b>Fuente</b>	RU-C07	<b>Prioridad</b>	Media
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	El usuario podrá destacar de forma individual cualquier marcador situado en el mapa. Un marcador destacado o resaltado predominará de forma visual sobre aquellos que no lo estén, y se representará de forma gráfica a través del color del usuario que lo haya destacado.		

TABLA 5.61: REQUISITO DE SOFTWARE RS-F19

RS-F19			
<b>Nombre</b>	Número de marcadores destacados por un usuario		
<b>Fuente</b>	RU-C07	<b>Prioridad</b>	Media
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	Cada usuario podrá destacar únicamente un marcador de forma simultánea.		

TABLA 5.62: REQUISITO DE SOFTWARE RS-F20

RS-F20			
<b>Nombre</b>	Número de usuarios que destacan un marcador		
<b>Fuente</b>	RU-C07	<b>Prioridad</b>	Media
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	Cada marcador estará destacado únicamente por un usuario de forma simultánea.		

TABLA 5.63: REQUISITO DE SOFTWARE RS-F21

RS-F21			
<b>Nombre</b>	Color asociado a usuario		
<b>Fuente</b>	RU-C07, RU-C09	<b>Prioridad</b>	Media
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	Cada usuario tendrá asociado un color, el cual indicará qué marcadores ha creado y qué marcador ha destacado.		

TABLA 5.64: REQUISITO DE SOFTWARE RS-F22

RS-F22			
<b>Nombre</b>	Color de destaque del marcador		
<b>Fuente</b>	RU-C07, RU-C09	<b>Prioridad</b>	Media
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	Cada marcador almacenará un número entero que representará el color con el que está destacado. Este color será el color asociado al usuario que lo ha destacado.		

TABLA 5.65: REQUISITO DE SOFTWARE RS-F23

RS-F23			
<b>Nombre</b>	Destacar de un marcador que ya está destacado		
<b>Fuente</b>	RU-C07	<b>Prioridad</b>	Media
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	Si un marcador ya está destacado por otro usuario, y otro usuario procede a destacarlo, el color de destaque del marcador será el del segundo. En otras palabras, un marcador está destacado por el último usuario que lo haya destacado.		

TABLA 5.66: REQUISITO DE SOFTWARE RS-F24

RS-F24			
<b>Nombre</b>	Eliminación de destaque de un marcador		
<b>Fuente</b>	RU-C08	<b>Prioridad</b>	Media
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	El usuario podrá revertir el estado de un marcador destacado anteriormente por él mismo, de tal forma que se muestre visualmente tal y como era antes de destacarse.		

TABLA 5.67: REQUISITO DE SOFTWARE RS-F25

RS-F25			
<b>Nombre</b>	Colores de los marcadores		
<b>Fuente</b>	RU-C09	<b>Prioridad</b>	Media
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	El color del marcador estará determinado por el usuario que lo haya creado. De esta forma, los marcadores creados por distintos usuarios serán de distinto color.		

TABLA 5.68: REQUISITO DE SOFTWARE RS-F26

RS-F26			
<b>Nombre</b>	Simultaneidad de colores en un marcador destacado		
<b>Fuente</b>	RU-C07, RU-C09	<b>Prioridad</b>	Media
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	El propietario de un marcador y el usuario que lo destaca no tiene por qué ser el mismo. El marcador debe mostrar los dos colores simultáneamente, indicando el propietario y el usuario que lo ha destacado.		

TABLA 5.69: REQUISITO DE SOFTWARE RS-F27

RS-F27			
<b>Nombre</b>	Eliminación de todos los marcadores propios		
<b>Fuente</b>	RU-C10	<b>Prioridad</b>	Baja
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	El usuario podrá eliminar con un comando todos los marcadores que haya creado anteriormente, es decir, todos los marcadores cuyo identificador del propietario coincidan con el identificador del usuario.		

## c) Requisitos de colaboración

TABLA 5.70: REQUISITO DE SOFTWARE RS-F28

RS-F28			
<b>Nombre</b>	Visualización marcadores de otros usuarios		
<b>Fuente</b>	RU-C11	<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	Cuando un usuario añade un marcador, el resto de los usuarios observarán el nuevo marcador en el mapa.		

TABLA 5.71: REQUISITO DE SOFTWARE RS-F29

RS-F29			
<b>Nombre</b>	Agrupación de usuarios en salas		
<b>Fuente</b>	RU-C17	<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	Una sala es el entorno en el cual un grupo de usuarios se comunican y colaboran interactuando con el mismo mapa. Un usuario solo puede pertenecer a una sala en un mismo instante, y cada sala será independiente de las demás.		

TABLA 5.72: REQUISITO DE SOFTWARE RS-F30

RS-F30			
<b>Nombre</b>	Límite de usuarios por sala		
<b>Fuente</b>	RU-C17	<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	El límite de usuarios por sala es de 8 jugadores.		

TABLA 5.73: REQUISITO DE SOFTWARE RS-F31

RS-F31			
<b>Nombre</b>	Sala única		
<b>Fuente</b>	RU-C17	<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	Sólo existirá una sala, a la cual entrarán los usuarios de forma automática.		

TABLA 5.74: REQUISITO DE SOFTWARE RS-F32

RS-F32			
<b>Nombre</b>	Adición de marcadores al entrar un usuario en sala		
<b>Fuente</b>	RU-C11	<b>Prioridad</b>	Media
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	Cuando un usuario entra en una sala, se añadirán a su mapa todos los marcadores previos a su entrada.		

TABLA 5.75: REQUISITO DE SOFTWARE RS-F33

RS-F33			
<b>Nombre</b>	Eliminación de marcadores de otros usuarios		
<b>Fuente</b>	RU-C12	<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	Cuando un usuario elimine un marcador, este desaparecerá en el mapa del resto de usuarios.		

TABLA 5.76: REQUISITO DE SOFTWARE RS-F34

RS-F34			
<b>Nombre</b>	Eliminación de marcadores al salir un usuario de la sala		
<b>Fuente</b>	RU-C12	<b>Prioridad</b>	Media
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	Cuando un usuario entra en una sala, el resto de los usuarios eliminarán todos los marcadores de ese usuario de sus respectivos mapas.		

TABLA 5.77: REQUISITO DE SOFTWARE RS-F35

RS-F35			
<b>Nombre</b>	Marcadores destacados por otros usuarios		
<b>Fuente</b>	RU-C13	<b>Prioridad</b>	Media
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	Cuando un usuario destaque un marcador, el resto de los usuarios percibirán visualmente el marcador resaltado. Cuando es eliminado este destaque, el resto de los usuarios también lo percibirá.		

TABLA 5.78: REQUISITO DE SOFTWARE RS-F36

RS-F36			
<b>Nombre</b>	Marcadores destacados al entrar un usuario en sala		
<b>Fuente</b>	RU-C13	<b>Prioridad</b>	Media
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	Cuando un usuario entra en una sala, los marcadores que estuvieran destacados se añadirán a su mapa con ese estado.		



TABLA 5.79: REQUISITO DE SOFTWARE RS-F37

RS-F37			
<b>Nombre</b>	Flechas que señalan marcadores destacados		
<b>Fuente</b>	RU-C13	<b>Prioridad</b>	Media
<b>Necesidad</b>	Deseable	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	Cuando un marcador destacado no sea visible en el mapa debido a acciones de desplazamiento o ampliación, se mostrarán unas flechas que señalarán la posición del marcador correspondiente.		

TABLA 5.80: REQUISITO DE SOFTWARE RS-F38

RS-F38			
<b>Nombre</b>	Color de las flechas que señalan marcadores destacados		
<b>Fuente</b>	RU-C13	<b>Prioridad</b>	Media
<b>Necesidad</b>	Deseable	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	El color de la flecha que señala a un marcador destacado será el color de destaque del marcador, es decir, el color asociado al usuario que ha destacado ese marcador.		

TABLA 5.81: REQUISITO DE SOFTWARE RS-F39

RS-F39			
<b>Nombre</b>	Número y posición de los marcadores		
<b>Fuente</b>	RU-C14	<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	Todos los usuarios observarán en sus respectivos mapas el mismo número de marcadores. Además, todos ellos señalarán los mismos lugares en el mapa.		

TABLA 5.82: REQUISITO DE SOFTWARE RS-F40

RS-F40			
<b>Nombre</b>	Chat de voz		
<b>Fuente</b>	RU-C15	<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	El usuario tendrá la capacidad de comunicarse con el resto de los usuarios a través de la utilización de un chat de voz.		

TABLA 5.83: REQUISITO DE SOFTWARE RS-F41

RS-F41			
<b>Nombre</b>	Silenciamiento del micrófono silenciado en chat de voz		
<b>Fuente</b>	RU-C15	<b>Prioridad</b>	Media
<b>Necesidad</b>	Deseable	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	El usuario podrá silenciar su micrófono, de tal forma que el resto de los usuarios no le escucharán.		

TABLA 5.84: REQUISITO DE SOFTWARE RS-F42

RS-F42			
<b>Nombre</b>	Activación de micrófono en chat de voz		
<b>Fuente</b>	RU-C15	<b>Prioridad</b>	Media
<b>Necesidad</b>	Deseable	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	El usuario podrá volver activar su micrófono, en caso de que este estuviera silenciado.		

TABLA 5.85: REQUISITO DE SOFTWARE RS-F43

RS-F43			
<b>Nombre</b>	Elementos gráficos del chat de voz		
<b>Fuente</b>	RU-C15	<b>Prioridad</b>	Media
<b>Necesidad</b>	Deseable	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	Cada usuario podrá saber a través de la observación de elementos gráficos cuándo habla el resto de los usuarios, así como cuándo están silenciados.		

## d) Otros requisitos

TABLA 5.86: REQUISITO DE SOFTWARE RS-F44

RS-F44			
<b>Nombre</b>	Elección de entorno virtual		
<b>Fuente</b>	RU-C16	<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	El usuario podrá elegir entre la inmersión en un entorno de realidad virtual o la interacción en un entorno real (de no realidad virtual).		

TABLA 5.87: REQUISITO DE SOFTWARE RS-F45

RS-F45			
<b>Nombre</b>	Alias del usuario		
<b>Fuente</b>	RU-C05	<b>Prioridad</b>	Media
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	Cada usuario podrá introducir un alias, en forma de cadena de caracteres. De esta forma, cada usuario podrá saber con qué usuarios se está comunicando.		

TABLA 5.88: REQUISITO DE SOFTWARE RS-F46

RS-F46			
<b>Nombre</b>	Lista de usuarios en una sala		
<b>Fuente</b>	RU-C09, RU-C15	<b>Prioridad</b>	Media
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	El sistema mostrará una lista de los usuarios que se encuentran en la sala, así como el alias y el color asociado de cada uno de ellos.		

TABLA 5.89: REQUISITO DE SOFTWARE RS-F47

RS-F47			
<b>Nombre</b>	Ocultación de lista de usuarios en sala		
<b>Fuente</b>	RU-C09, RU-C15	<b>Prioridad</b>	Media
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	El usuario tendrá la capacidad de ocultar la lista de usuarios que están en la misma sala.		

TABLA 5.90: REQUISITO DE SOFTWARE RS-F48

RS-F48			
<b>Nombre</b>	Mostrado de lista de usuarios en sala		
<b>Fuente</b>	RU-C09, RU-C15	<b>Prioridad</b>	Media
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	El usuario tendrá la capacidad de volver a mostrar la lista de usuarios que están en la misma sala, en caso de que esta estuviera oculta.		

### 5.4.2. Requisitos de rendimiento

TABLA 5.91: REQUISITO DE SOFTWARE RS-RE01

RS-RE01			
<b>Nombre</b>	Respuesta de la aplicación		
<b>Fuente</b>	RU-R11	<b>Prioridad</b>	Media
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Baja
<b>Descripción</b>	El tiempo de respuesta de la aplicación debe ser menor a 200 ms.		

### 5.4.3. Requisitos operacionales

TABLA 5.92: REQUISITO DE SOFTWARE RS-OP01

RS-OP01			
<b>Nombre</b>	Plataforma de desarrollo		
<b>Fuente</b>	RU-R01	<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	El desarrollo de la aplicación se realizará utilizando el motor de juego Unity3D, preferiblemente la versión 5.4.0f3.		

TABLA 5.93: REQUISITO DE SOFTWARE RS-OP02

RS-OP02			
<b>Nombre</b>	Aplicación de realidad virtual		
<b>Fuente</b>	RU-R02	<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	La aplicación debe ser compatible con la tecnología de realidad virtual, es decir, debe poder observarse el entorno generado utilizando un HMD de realidad virtual.		

TABLA 5.94: REQUISITO DE SOFTWARE RS-OP03

<b>RS-OP03</b>			
<b>Nombre</b>	Compatibilidad con Oculus		
<b>Fuente</b>	RU-R03	<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	La aplicación debe ser compatible con el HMD Oculus Rift CV1.		

TABLA 5.95: REQUISITO DE SOFTWARE RS-OP04

<b>RS-OP04</b>			
<b>Nombre</b>	Controles de realidad virtual		
<b>Fuente</b>	RU-R05	<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	La interacción entre el usuario y la aplicación en el entorno de realidad virtual debe realizarse utilizando los controles Oculus Touch.		

TABLA 5.96: REQUISITO DE SOFTWARE RS-OP05

<b>RS-OP05</b>			
<b>Nombre</b>	Controles de no realidad virtual		
<b>Fuente</b>	RU-R06	<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	La interacción entre el usuario y la aplicación en el entorno que no es de realidad virtual debe realizarse utilizando ratón y teclado.		

TABLA 5.97: REQUISITO DE SOFTWARE RS-OP06

RS-OP06			
<b>Nombre</b>	Desarrollo de realidad virtual		
<b>Fuente</b>	RU-R08	<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	El desarrollo de realidad virtual debe realizarse en un ordenador que posea las especificaciones recomendadas por Oculus o superiores.		

#### 5.4.4. Requisitos de calidad

TABLA 5.98: REQUISITO DE SOFTWARE RS-CA01

RS-CA01			
<b>Nombre</b>	Adecuación de respuesta		
<b>Fuente</b>	-	<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	La respuesta del sistema ante una determinada operación se adecúa a la especificada en el plan de pruebas.		

TABLA 5.99: REQUISITO DE SOFTWARE RS-CA02

RS-CA02			
<b>Nombre</b>	Adecuación de requisitos de usuario		
<b>Fuente</b>	-	<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	El sistema posee todas las funcionalidades especificadas por el cliente.		

### 5.4.5. Requisitos de interfaz

TABLA 5.100: REQUISITO DE SOFTWARE RS-I01

RS-I01			
<b>Nombre</b>	Servidor		
<b>Fuente</b>	-	<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	El sistema debe comunicarse con un servidor encargado de controlar el tráfico de red entre los usuarios.		

TABLA 5.101: REQUISITO DE SOFTWARE RS-I02

RS-I02			
<b>Nombre</b>	Extensión a entornos de no realidad virtual		
<b>Fuente</b>	RU-R04	<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	Debe ser posible la ejecución de la aplicación en entornos que no sean de realidad virtual, es decir, debe poder ejecutarse en monitores o pantallas.		

TABLA 5.102: REQUISITO DE SOFTWARE RS-I03

RS-I03			
<b>Nombre</b>	Acceso a internet		
<b>Fuente</b>	RU-R09	<b>Prioridad</b>	Alta
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	La aplicación debe tener acceso a internet para poder llevar a cabo la colaboración.		



#### 5.4.6. Requisitos de usabilidad

TABLA 5.103: REQUISITO DE SOFTWARE RS-U01

RS-U01			
<b>Nombre</b>	Visualización de la información		
<b>Fuente</b>	RU-R10	<b>Prioridad</b>	Media
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	La información mostrada en la ejecución de la aplicación debe ser nítida y legible en caso de tratarse de texto.		

TABLA 5.104: REQUISITO DE SOFTWARE RS-U02

RS-U02			
<b>Nombre</b>	Intuitividad		
<b>Fuente</b>	RU-R12	<b>Prioridad</b>	Media
<b>Necesidad</b>	Esencial	<b>Verificabilidad</b>	Alta
<b>Descripción</b>	La aplicación debe ser intuitiva, facilitando el aprendizaje de los controles. Además, los elementos gráficos mostrados, como es el caso de los iconos, deben expresar correctamente su significado.		

### 5.5. Análisis de clases

En este apartado, se realiza un análisis de clases, enfocado en conocer la información con la que se va a trabajar posteriormente, en la fase de implementación.

Para ello, se ha construido un diagrama de clases, el cual mostrará las principales entidades y los datos que estas necesitan almacenar. Este diagrama es el que se muestra a continuación:

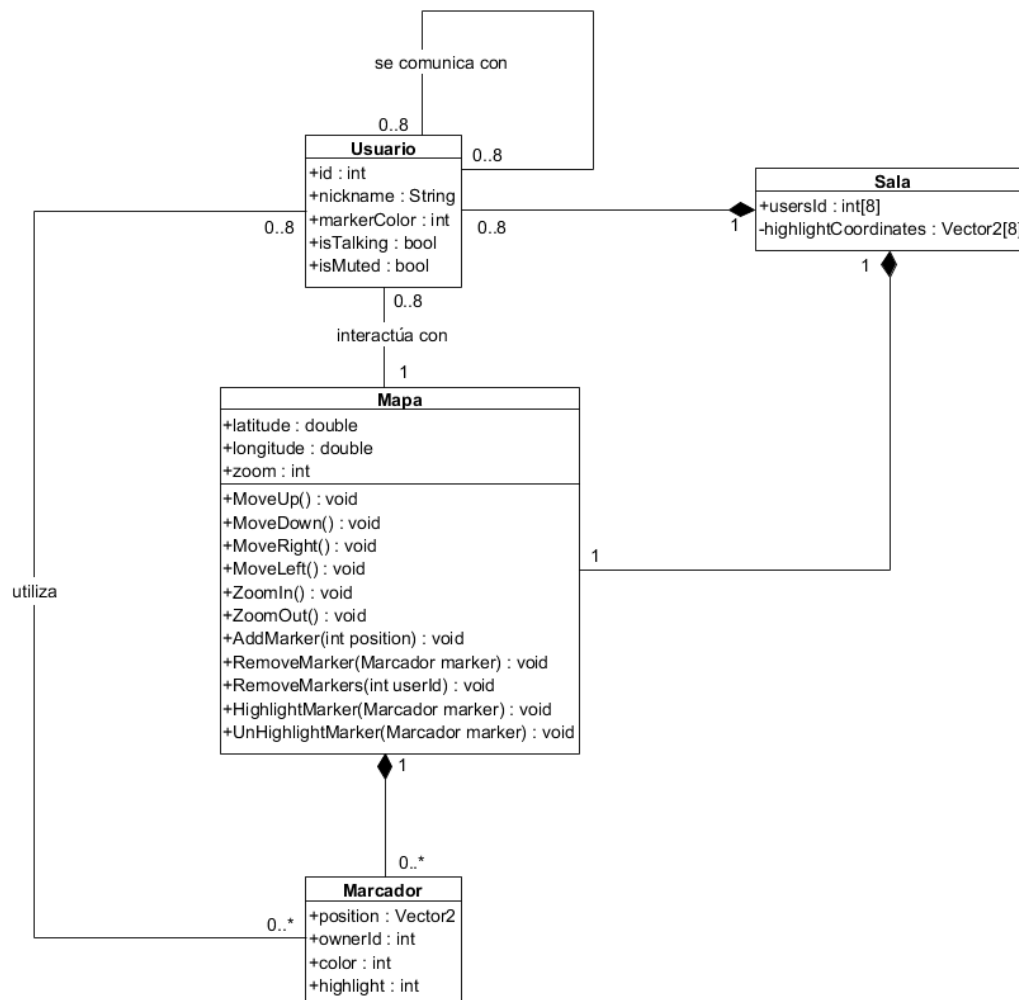


Fig. 5.2. Diagrama de clases

Llegados a este punto, se especifican las responsabilidades y los atributos de cada una de las entidades mostradas en el diagrama de clases. Para ello, se utiliza una tabla como la siguiente:

TABLA 5.105: PLANTILLA DE IDENTIFICACIÓN DE CLASES

Nombre de la clase	
<b>Responsabilidades</b>	
<b>Atributos</b>	
<b>Operaciones</b>	Opcional.

La tabla contiene los siguientes atributos:

- **Nombre de la clase.** Es el nombre de la clase que se va a estudiar.
- **Responsabilidades.** Se trata de las obligaciones y la utilidad de la clase para el proyecto.
- **Atributos.** Se enumera y define la información que conforman la clase, indicando los tipos de datos. Se consideran los atributos más importantes para la resolución del problema.
- **Operaciones.** Se enumeran y describen las operaciones más importantes.

A continuación, se detallan las clases de entidad mencionadas:

TABLA 5.106: CLASE SALA

Sala	
<b>Responsabilidades</b>	Gestiona la información común a la sala en la que se encuentran los usuarios.
<b>Atributos</b>	<p><b><i>UsersId (int [8])</i></b>. Array que indica el color asociado a cada usuario de la sala. Evita tener que obtener esta información recorriendo todos los usuarios de la sala.</p> <p><b><i>HighlightCoordinates (Vector2 [8])</i></b>. Array que indica la posición del marcador destacado por cada usuario. Esta información es útil para el mostrado de flechas cuando el marcador destacado se encuentra fuera del rango de visión.</p>

TABLA 5.107: CLASE USUARIO

Usuario	
<b>Responsabilidades</b>	Gestiona la información de los usuarios de la aplicación. Esta información es útil de cara a la coherencia y la consistencia de los datos en la colaboración.
<b>Atributos</b>	<p><b>Id (int).</b> Identificador que distingue al usuario de forma unívoca.</p> <p><b>Nickname (string).</b> Alias o nombre del usuario.</p> <p><b>MarkerColor (int).</b> Número entero que determina el color asociado al usuario. Este color se utilizará para crear y destacar marcadores.</p> <p><b>IsTalking (bool).</b> Indica si el usuario está transmitiendo datos de audio o no.</p> <p><b>IsMuted (bool).</b> Indica si el usuario tiene o no el micrófono silenciado.</p>

TABLA 5.108: CLASE MARCADOR

Marcador	
<b>Responsabilidades</b>	Gestiona la información relativa al mostrado del mapa.
<b>Atributos</b>	<p><b>Position (Vector2).</b> Indica la posición del marcador en el mapa en términos de latitud y longitud.</p> <p><b>OwnerId (int).</b> Identificador del usuario que ha creado el marcador, el usuario propietario.</p> <p><b>Color (int).</b> Número entero que representa el color asociado al usuario propietario, es decir, el color en el que se muestra el marcador.</p> <p><b>Highlight (int).</b> Número entero que representa el color asociado al usuario que ha destacado el marcador, es decir, el color de destaque del marcador.</p>

TABLA 5.109: CLASE MAPA

Mapa	
<b>Responsabilidades</b>	Gestiona la información relativa al mostrado del mapa. Ofrece funciones para gestionar el movimiento del mapa y sus marcadores.
<b>Atributos</b>	<p><b><i>Latitude (double)</i></b>. Indica la posición del centro del plano observable del mapa en términos de latitud.</p> <p><b><i>Longitude (double)</i></b>. Indica la posición del centro del plano observable del mapa en términos de longitud.</p> <p><b><i>Zoom (int)</i></b>. Nivel de <i>zoom</i> actual.</p>
<b>Operaciones</b>	<p><b><i>MoveUp ()</i></b>. Desplazamiento hacia arriba.</p> <p><b><i>MoveDown ()</i></b>. Desplazamiento hacia abajo.</p> <p><b><i>MoveRight ()</i></b>. Desplazamiento hacia la derecha.</p> <p><b><i>MoveLeft ()</i></b>. Desplazamiento hacia la izquierda.</p> <p><b><i>ZoomIn ()</i></b>. Ampliación del mapa.</p> <p><b><i>ZoomOut ()</i></b>. Disminución de la ampliación del mapa.</p> <p><b><i>AddMarker (int position)</i></b>. Añade un marcador en la posición indicada.</p> <p><b><i>RemoveMarker (Marcador marker)</i></b>. Elimina el marcador recibido por parámetro.</p> <p><b><i>RemoveMarkers (int userId)</i></b>. Elimina los marcadores creados por un usuario.</p> <p><b><i>HighlightMarker (Marcador marker)</i></b>. Destaca un marcador.</p> <p><b><i>UnHighlightMarker (Marcador marker)</i></b>. Elimina el destaque de un marcador.</p>

## 5.6. Trazabilidad

En este apartado se verifica la consistencia entre las diferentes fases del proceso de análisis, a través de la creación de matrices de trazabilidad entre los requisitos de software y el resto de los elementos.

### 5.6.1. Requisitos de Usuario – Requisitos de Software

La siguiente tabla representa la matriz de trazabilidad entre requisitos de capacidad y requisitos funcionales. De esta forma, se verifica que todo requisito de software tenga un origen, al igual que todo requisito de usuario se haya tenido en cuenta al generar los requisitos funcionales. Se marca con una 'x' en caso de que el requisito de software tenga origen en el requisito de usuario.

TABLA 5.110: TRAZABILIDAD RU – RS (I)

RU RS	RU-C01	RU-C02	RU-C03	RU-C04	RU-C05	RU-C06	RU-C07	RU-C08	RU-C09	RU-C10	RU-C11	RU-C12	RU-C13	RU-C14	RU-C15	RU-C16	RU-C17
RS-F01	x																
RS-F02		x															
RS-F03			x														
RS-F04		x	x														
RS-F05		x															
RS-F06			x														
RS-F07		x	x														
RS-F08				x													
RS-F09	x																
RS-F10					x												
RS-F11					x												
RS-F12					x												
RS-F13					x												
RS-F14						x											
RS-F15						x											
RS-F16						x											
RS-F17						x											
RS-F18							x										
RS-F19							x										
RS-F20							x										
RS-F21							x		x								

TABLA 5.111: TRAZABILIDAD RU – RS (II)

RU RS	RU-C01	RU-C02	RU-C03	RU-C04	RU-C05	RU-C06	RU-C07	RU-C08	RU-C09	RU-C10	RU-C11	RU-C12	RU-C13	RU-C14	RU-C15	RU-C16	RU-C17
RS-F22							x		x								
RS-F23							x										
RS-F24								x									
RS-F25									x								
RS-F26							x		x								
RS-F27										x							
RS-F28											x						
RS-F29																	x
RS-F30																	x
RS-F31																	x
RS-F32											x						
RS-F33												x					
RS-F34												x					
RS-F35													x				
RS-F36													x				
RS-F37													x				
RS-F38													x				
RS-F39														x			
RS-F40															x		
RS-F41															x		
RS-F42															x		
RS-F43															x		
RS-F44																x	
RS-F45					x												
RS-F46									x						x		
RS-F47									x						x		
RS-F48									x						x		

### 5.6.2. Casos de Uso – Requisitos de Software

La siguiente tabla representa la matriz de trazabilidad entre casos de uso y requisitos de software. De esta forma, se verifica su consistencia, teniendo en cuenta que ambos proceden de los requisitos de usuario.

TABLA 5.112: TRAZABILIDAD CU – RS (I)

CU RS	CU-01	CU-02	CU-03	CU-04	CU-05	CU-06	CU-07	CU-08	CU-09	CU-10
RS-F01	x	x	x	x	x	x	x	x		
RS-F02	x									
RS-F03		x								
RS-F04	x	x								
RS-F05	x									
RS-F06		x								
RS-F07	x	x								
RS-F08			x							
RS-F09	x	x	x	x	x	x	x	x		
RS-F10					x					
RS-F11					x					
RS-F12					x					
RS-F13					x					
RS-F14						x				
RS-F15						x				
RS-F16						x				
RS-F17						x				
RS-F18							x			
RS-F19							x			
RS-F20							x			
RS-F21					x		x			
RS-F22					x		x			
RS-F23							x			
RS-F24								x		
RS-F25					x		x			
RS-F26					x		x			
RS-F27				x						



TABLA 5.113: TRAZABILIDAD CU – RS (II)

CU RS	CU-01	CU-02	CU-03	CU-04	CU-05	CU-06	CU-07	CU-08	CU-09	CU-10
RS-F28					x					
RS-F29	x	x	x	x	x	x	x	x	x	x
RS-F30	x	x	x	x	x	x	x	x	x	x
RS-F31	x	x	x	x	x	x	x	x	x	x
RS-F32					x					
RS-F33				x		x				
RS-F34				x		x				
RS-F35							x	x		
RS-F36							x	x		
RS-F37							x	x		
RS-F38							x	x		
RS-F39				x	x	x				
RS-F40									x	x
RS-F41									x	x
RS-F42									x	x
RS-F43									x	x
RS-F44	x	x	x	x	x	x	x	x	x	x
RS-F45					x					
RS-F46					x				x	x
RS-F47					x				x	x
RS-F48					x				x	x

### 5.6.3. Clases – Requisitos de Software

La siguiente tabla representa la matriz de trazabilidad entre clases y requisitos de software. De esta forma, se verifica si se tuvieron en cuenta todos los requisitos de software a la hora de realizar el análisis de clases.

TABLA 5.114: TRAZABILIDAD CLASES – RS (I)

Clase RS	Sala	Usuario	Marcador	Mapa
RS-F01				x
RS-F02				x
RS-F03				x
RS-F04				x
RS-F05				x
RS-F06				x
RS-F07				x
RS-F08				x
RS-F09				x
RS-F10			x	x
RS-F11		x		
RS-F12		x	x	
RS-F13			x	
RS-F14		x	x	x
RS-F15		x	x	x
RS-F16			x	x
RS-F17			x	x
RS-F18	x	x	x	x
RS-F19	x	x	x	x
RS-F20		x	x	x
RS-F21		x		
RS-F22	x		x	
RS-F23	x	x	x	x
RS-F24	x	x	x	x
RS-F25		x	x	x
RS-F26	x		x	x
RS-F27		x	x	x

TABLA 5.115: TRAZABILIDAD CLASES – RS (II)

Clase RS	Sala	Usuario	Marcador	Mapa
RS-F28			x	x
RS-F29	x			
RS-F30	x			
RS-F31	x			
RS-F32			x	x
RS-F33			x	x
RS-F34			x	x
RS-F35			x	x
RS-F36			x	x
RS-F37	x			x
RS-F38			x	x
RS-F39			x	x
RS-F40		x		
RS-F41		x		
RS-F42		x		
RS-F43		x		
RS-F44				x
RS-F45		x		
RS-F46		x		
RS-F47		x		
RS-F48		x		

## 6. DISEÑO

Tras la fase de análisis, se construirá el diseño de la arquitectura del sistema. Este diseño se plasmará en el documento en forma de diagramas y tablas.

En primer lugar, se define la arquitectura del sistema, dividiéndolo en subsistemas. Estos subsistemas se dividen a su vez en componentes. A continuación, se realizará una descripción de cada componente, indicando su función en el sistema y su relación con el resto de los componentes. Para verificar que el diseño del sistema abarca todos los requisitos especificados en la fase de análisis, se creará una matriz de trazabilidad entre componentes y requisitos de software.

Por último, se incluye un apartado en el que se detallan las decisiones de diseño tomadas para formar los principales elementos gráficos de la aplicación.

### 6.1. Arquitectura

La arquitectura del sistema sigue el modelo de aplicación en tres capas, el cual divide el sistema en tres niveles que se comunican entre sí:

- **Capa de Presentación.** Es el nivel encargado de comunicarse con el usuario, a través de una interfaz gráfica.
- **Capa de Negocio.** Se trata del nivel que contiene la lógica de la aplicación. Procesa las peticiones de la capa de presentación y devuelve el resultado.
- **Capa de Datos.** Gestiona el almacenamiento y el acceso a la información que solicita la capa de negocio. Esta capa no se da en el sistema, ya que no se almacenan datos de forma persistente. No obstante, podrá ser utilizada en el futuro, si es necesario el almacenamiento de datos.

De cara al diseño de la arquitectura del sistema, es necesaria la división en subsistemas. Cada subsistema pertenecerá a una de las capas descritas anteriormente. Estos subsistemas se dividen en componentes, que representan cada una de las funciones que requiere la aplicación para su correcto funcionamiento.

En este caso, debido a la cantidad de funciones que poseerá la aplicación, se muestra un modelo simplificado, en el que cada componente representa un conjunto de funciones que guardan relación entre sí. Asimismo, únicamente se tendrán en cuenta aquellos componentes en los que se dará la implementación, además de las librerías utilizadas. Por tanto, se asume la existencia de los componentes que ofrece Unity como motor de juego, pero no se tienen en cuenta en el diseño de la arquitectura del sistema.

Los diferentes subsistemas en los que se ha dividido la aplicación son los siguientes:

- **Subsistema de elementos gráficos.** Subsistema de la capa de presentación que engloba todos los elementos de la interfaz de usuario.
- **Subsistema de control de juego.** Este subsistema pertenece a la capa de presentación. Engloba aquellos componentes que ofrecen los controles al usuario para ejecutar las diferentes acciones.

- **Subsistema de interacción con mapas y uso de marcadores.** Subsistema de la capa de negocio que gestiona la navegación a través del mapa, así como el manejo de marcadores.
- **Subsistema de gestión de elementos gráficos.** Este subsistema también pertenece a la capa de negocio, y se encarga de la administración de los elementos gráficos en cuanto a visibilidad o posición en el entorno virtual.
- **Subsistema de colaboración.** Se trata de otro subsistema de la capa de negocio cuyos componentes se ocupan de dirigir la comunicación entre los usuarios, tanto a nivel de software como a nivel físico.

A partir de la creación de estos subsistemas, se ha generado el diagrama de componentes de la aplicación, mostrado en la siguiente figura:

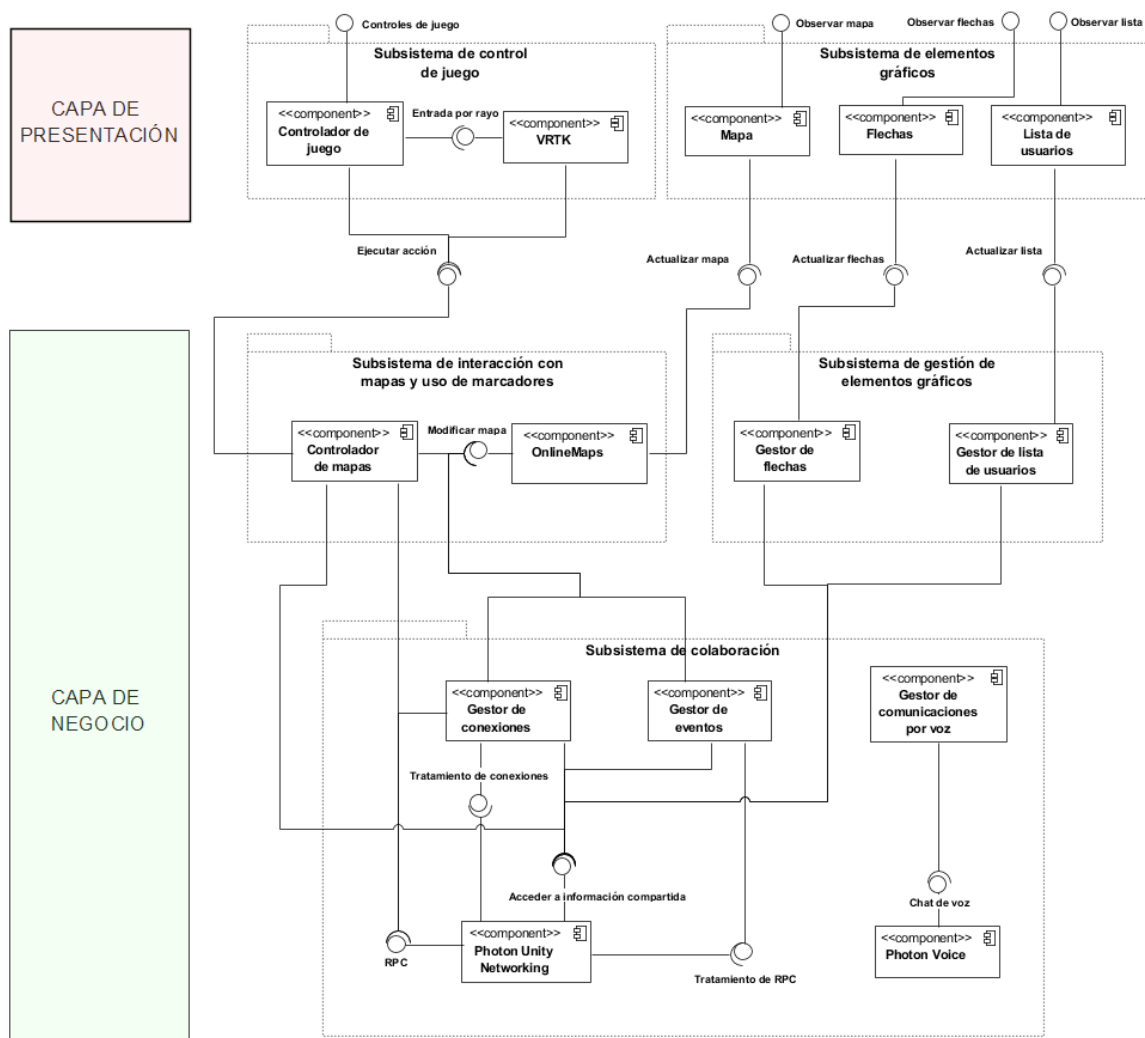


Fig. 6.1. Diagrama de componentes

## 6.2. Descripción de los componentes

En este apartado se describe cada uno de los componentes que aparecen en el diagrama anterior. Estos componentes se definen a través de un conjunto de propiedades:

- **Identificador.** Cada componente posee un identificador único. La nomenclatura utilizada es del formato CO-XX, donde CO indica que se trata de un componente, y XX representa un número entero de dos cifras que comienza por el número uno (01) y se va incrementando de unidad en unidad.
- **Nombre.** Breve resumen de la función del componente o nombre de la librería.
- **Subsistema.** Nombre del subsistema al que pertenece.
- **Función.** Se trata de las tareas que realiza el componente.
- **Subordinados.** Indica aquellos componentes que dependen de este, es decir, los componentes que utilizan sus interfaces.
- **Dependencias.** Componentes de los que depende, debido a que utiliza las interfaces que estos ofrecen.

No se incluyen en las tablas los requisitos de los que procede cada componente, con el fin de obtener tablas menos recargadas. Esto se debe a que el número de requisitos de software es significativamente superior al número de componentes. Para comprobar si todos los requisitos han sido cubiertos, se observará la matriz de trazabilidad del apartado 6.3.

Dadas estas propiedades, se utilizarán tablas que seguirán la siguiente plantilla:

TABLA 6.1: PLANTILLA DE COMPONENTES

CO-XX	
<b>Nombre</b>	
<b>Subsistema</b>	
<b>Función</b>	
<b>Subordinados</b>	
<b>Dependencias</b>	

A continuación, se realiza una descripción de cada uno de los componentes de la aplicación:

TABLA 6.2: COMPONENTE CO-01

<b>CO-01</b>	
<b>Nombre</b>	Controlador de juego
<b>Subsistema</b>	Subsistema de control de juego
<b>Función</b>	Recibe los comandos por parte del usuario y llama a las funciones de negocio correspondientes.
<b>Subordinados</b>	Ninguno
<b>Dependencias</b>	CO-02, CO-06

TABLA 6.3: COMPONENTE CO-02

<b>CO-02</b>	
<b>Nombre</b>	VRTK
<b>Subsistema</b>	Subsistema de control de juego
<b>Función</b>	Librería que ofrece herramientas para entornos de realidad virtual. Se encarga de controlar el método de entrada a través de rayo.
<b>Subordinados</b>	CO-01
<b>Dependencias</b>	CO-06

TABLA 6.4: COMPONENTE CO-03

<b>CO-03</b>	
<b>Nombre</b>	Mapa
<b>Subsistema</b>	Subsistema de elementos gráficos
<b>Función</b>	Es el mapa que observa el usuario durante la ejecución de la aplicación. Sobre él se colocan los marcadores.
<b>Subordinados</b>	Ninguno
<b>Dependencias</b>	CO-07

TABLA 6.5: COMPONENTE CO-04

<b>CO-04</b>	
<b>Nombre</b>	Flechas
<b>Subsistema</b>	Subsistema de elementos gráficos
<b>Función</b>	Estas flechas indican la posición de un marcador destacado cuando este se encuentra fuera del rango de visión.
<b>Subordinados</b>	Ninguno
<b>Dependencias</b>	CO-08

TABLA 6.6: COMPONENTE CO-05

<b>CO-05</b>	
<b>Nombre</b>	Lista de usuarios
<b>Subsistema</b>	Subsistema de elementos gráficos
<b>Función</b>	Indica el nombre y color de todos los usuarios de la sala, además de un icono que indica si está transmitiendo audio o si tiene el micrófono silenciado.
<b>Subordinados</b>	Ninguno
<b>Dependencias</b>	CO-09



TABLA 6.7: COMPONENTE CO-06

<b>CO-06</b>	
<b>Nombre</b>	Controlador de mapa
<b>Subsistema</b>	Subsistema de interacción con mapas y uso de marcadores
<b>Función</b>	Contiene todas las funciones de alto nivel que trabajan con el mapa y sus marcadores.
<b>Subordinados</b>	CO-01, CO-02, CO-03
<b>Dependencias</b>	CO-07, CO-12

TABLA 6.8: COMPONENTE CO-07

<b>CO-07</b>	
<b>Nombre</b>	OnlineMaps
<b>Subsistema</b>	Subsistema de interacción con mapas y uso de marcadores
<b>Función</b>	Librería que contiene las funciones de bajo nivel que trabajan con el mapa y sus marcadores, a excepción del destaque de marcadores.
<b>Subordinados</b>	CO-06
<b>Dependencias</b>	Ninguna

TABLA 6.9: COMPONENTE CO-08

<b>CO-08</b>	
<b>Nombre</b>	Gestión de flechas
<b>Subsistema</b>	Subsistema de gestión de elementos gráficos
<b>Función</b>	Controla el mostrado de las flechas que apuntan a los marcadores destacados que se encuentren fuera del rango de visión.
<b>Subordinados</b>	CO-04
<b>Dependencias</b>	CO-12

TABLA 6.10: COMPONENTE CO-09

<b>CO-09</b>	
<b>Nombre</b>	Gestión de lista de usuarios
<b>Subsistema</b>	Subsistema de gestión de elementos gráficos
<b>Función</b>	Controla el mostrado de la lista de usuarios, así como la información que debe mostrar en cada momento.
<b>Subordinados</b>	CO-05
<b>Dependencias</b>	CO-12, CO-13

TABLA 6.11: COMPONENTE CO-10

<b>CO-10</b>	
<b>Nombre</b>	Gestor de conexiones
<b>Subsistema</b>	Subsistema de colaboración
<b>Función</b>	Engloba todas las rutinas relacionadas con la conexión y la desconexión de cada uno de los usuarios.
<b>Subordinados</b>	CO-12
<b>Dependencias</b>	CO-12

TABLA 6.12: COMPONENTE CO-11

<b>CO-11</b>	
<b>Nombre</b>	Gestor de eventos
<b>Subsistema</b>	Subsistema de colaboración
<b>Función</b>	Se encarga de mantener la coherencia y la consistencia de los mapas de todos los usuarios. Gestiona el tratamiento de eventos relacionados con acciones de otros usuarios.
<b>Subordinados</b>	CO-12
<b>Dependencias</b>	CO-06

TABLA 6.13: COMPONENTE CO-12

<b>CO-12</b>	
<b>Nombre</b>	Photon Unity Networking
<b>Subsistema</b>	Subsistema de colaboración
<b>Función</b>	Librería utilizada para proporcionar la colaboración entre los usuarios. Este componente se conecta al servidor que gestiona las comunicaciones internas entre los usuarios, además de proporcionar a cada uno de ellos la información asociada a la sala y al resto de los usuarios.
<b>Subordinados</b>	CO-06, CO-08, CO-09, CO-10
<b>Dependencias</b>	CO-10, CO-11

TABLA 6.14: COMPONENTE CO-13

<b>CO-13</b>	
<b>Nombre</b>	Gestor de comunicaciones por voz
<b>Subsistema</b>	Subsistema de colaboración
<b>Función</b>	Se encarga de proporcionar el chat de voz, gestionando las transmisiones de audio de cada usuario.
<b>Subordinados</b>	CO-09
<b>Dependencias</b>	CO-14

TABLA 6.15: COMPONENTE CO-14

<b>CO-14</b>	
<b>Nombre</b>	Photon Voice
<b>Subsistema</b>	Subsistema de colaboración
<b>Función</b>	Librería que contiene las funciones de bajo nivel para realizar las comunicaciones por voz.
<b>Subordinados</b>	CO-13
<b>Dependencias</b>	Ninguna

### 6.3. Trazabilidad Componentes – Requisitos de Software

La siguiente tabla representa la matriz de trazabilidad entre componentes y requisitos de software. De esta forma, se verifica que todo componente tiene su origen en algún requisito de software y, a su vez, todo requisito de software origina algún componente.

TABLA 6.16: TRAZABILIDAD CO – RS (I)

CO RS	CO-01	CO-02	CO-03	CO-04	CO-05	CO-06	CO-07	CO-08	CO-09	CO-10	CO-11	CO-12	CO-13	CO-14
RS-F01			X				X							
RS-F02	X		X			X	X							
RS-F03	X		X			X	X							
RS-F04			X			X	X							
RS-F05						X								
RS-F06						X								
RS-F07						X								
RS-F08	X		X			X	X							
RS-F09	X		X			X								
RS-F10	X	X	X			X	X					X		
RS-F11						X						X		
RS-F12						X								
RS-F13						X	X							
RS-F14	X	X	X			X	X					X		
RS-F15						X						X		
RS-F16	X	X	X			X								
RS-F17	X	X	X			X								
RS-F18	X	X	X			X	X					X		
RS-F19						X								
RS-F20						X								
RS-F21			X			X				X		X		
RS-F22			X			X								
RS-F23			X			X	X							
RS-F24	X	X	X			X	X					X		
RS-F25			X			X	X					X		
RS-F26			X			X	X							
RS-F27	X		X			X	X					X		

TABLA 6.17: TRAZABILIDAD CO – RS (II)

CO RS	CO-01	CO-02	CO-03	CO-04	CO-05	CO-06	CO-07	CO-08	CO-09	CO-10	CO-11	CO-12	CO-13	CO-14
RS-F28						X					X	X		
RS-F29										X		X		
RS-F30										X		X		
RS-F31						X				X		X		
RS-F32			X			X				X	X	X		
RS-F33						X					X	X		
RS-F34			X			X				X		X		
RS-F35						X					X	X		
RS-F36			X							X	X	X		
RS-F37				X				X				X		
RS-F38				X				X				X		
RS-F39						X								
RS-F40													X	X
RS-F41	X												X	X
RS-F42	X												X	X
RS-F43	X				X				X					
RS-F44						X								
RS-F45						X						X		
RS-F46					X				X			X		
RS-F47	X				X				X					
RS-F48	X				X				X					

## 6.4. Elementos gráficos y controles de juego

En este apartado, se detallan las decisiones de diseño tomadas respecto a los elementos gráficos que se muestran en la aplicación, así como los controles de juego.

### 6.4.1. Elementos gráficos

- **Mapa.** Es el elemento gráfico principal. Para su creación, han seguido las recomendaciones de la librería *OnlineMaps*. Se trata de un plano vertical, y es el único elemento que observará la cámara. Se utilizan dos cámaras diferentes para los entornos con y sin realidad virtual, debido a que el campo de visión en realidad virtual es mayor.
- **Cursor y retículo.** Se ha modificado el cursor del ratón, de tal forma que coincide con el retículo (centro de la mirada) en realidad virtual. De esta forma, en ambos entornos, se utiliza el mismo elemento para señalar el mapa. En el caso del entorno de realidad virtual, también se permite la entrada mediante un rayo con origen en el control Oculus Touch derecho.
- **Textura de marcadores.** La textura utilizada para la creación de marcadores se basa en la que ofrece la librería *OnlineMaps*. Se ha añadido un borde interior de color negro, con el fin de que sea más distinguible en el mapa. Para diferenciar los marcadores que añaden diferentes usuarios, se han creado ocho marcadores de diferentes colores, además de un noveno marcador de color gris, que indicará al usuario sobre qué marcador realizará la acción correspondiente.

Los seis primeros colores corresponden con los colores primarios y secundarios del modelo aditivo de colores: rojo, verde, azul, cian, magenta y amarillo. Los dos colores restantes serán colores terciarios, en este caso el naranja y el morado. Además, la saturación de estos será la mayor posible, favoreciendo que los marcadores sean distinguibles en el mapa.

En la figura 6.2, se muestran los marcadores de diferentes colores y su textura de destaque. Asimismo, se indica el número asociado a cada color.



Fig. 6.2. Texturas de marcadores

- **Textura de destaque de marcadores.** Se utilizan los mismos colores en las texturas para destacar los marcadores. Cuando un usuario destaca un marcador, este es rodeado por un círculo de su color asociado. No se crean texturas de las diferentes combinaciones entre marcador y textura de destaque, sino que se combinan ambas en tiempo de ejecución. Asimismo, se duplica el tamaño del marcador destacado para que llame la atención del resto de usuarios. La figura 6.3 señala la diferencia entre un marcador destacado y otro sin destacar a nivel gráfico.



Fig. 6.3. Comparación entre un marcador destacado y otro sin destacar

- **Flechas.** Las flechas apuntan aquellos marcadores destacados que no se observan en el plano vertical del mapa. Generalmente, su aparición está asociada a un nivel de *zoom* alto. Se utiliza el mismo conjunto de ocho colores descrito anteriormente. En este caso, únicamente se ha creado una textura base de color blanco (figura 6.4) y, posteriormente, se añade el color deseado a través del editor de Unity. Esta textura se replica ocho veces y forma parte de un *canvas* situado delante del mapa. Además, las flechas parpadean para llamar la atención del usuario.



Fig. 6.4. Textura base de las flechas

- **Lista de usuarios.** En la lista de usuarios, se indicará información útil para la colaboración. Se mostrará un marcador que indica el color asociado al usuario y su alias o nombre de usuario. En caso de que el usuario esté transmitiendo voz, se indica a través del icono de un altavoz. Asimismo, se muestra el icono de un micrófono tachado si este está silenciado. La figura 6.5 es un ejemplo de lista de usuarios en la que aparecen todos los elementos descritos.

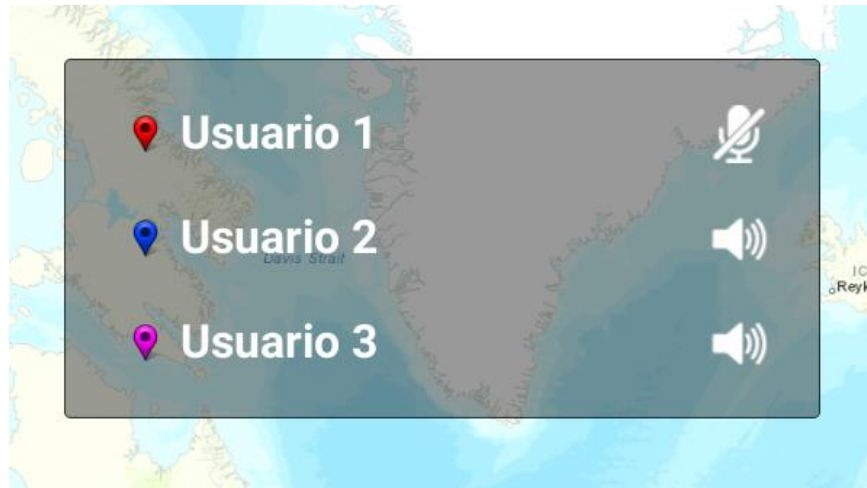


Fig. 6.5. Lista de usuarios



### 6.4.2. Controles de juego

Para el entorno sin realidad virtual, la interacción se realizará a través de un teclado y un ratón, de tal forma que son similares a los controles de videojuegos en PC. Los controles de juego se muestran a través de la tabla 6.18 y la figura 6.6:

TABLA 6.18: CONTROLES DE JUEGO EN ENTORNO SIN REALIDAD VIRTUAL

<b>Función</b>	<b>Controles de ratón o teclado</b>
Desplazar mapa hacia arriba	W o ↑
Desplazar mapa hacia abajo	S o ↓
Desplazar mapa hacia la derecha	D o →
Desplazar mapa hacia la izquierda	A o ←
<i>Zoom in</i>	Girar rueda del ratón hacia delante ( <i>scroll up</i> )
<i>Zoom out</i>	Girar rueda del ratón hacia atrás ( <i>scroll down</i> )
Añadir marcador	Botón izquierdo del ratón
Eliminar marcador	Botón derecho del ratón
Eliminar todos los marcadores propios	Mantener pulsado botón derecho del ratón durante un segundo
Destacar marcador	Botón central del ratón sobre un marcador que no esté destacado por el usuario
Eliminar destaque de marcador	Botón central del ratón sobre un marcador que sí esté destacado por el usuario
Mostrar / ocultar lista de usuarios	TAB
Activar / silenciar micrófono	M
Salir de la aplicación	ESC

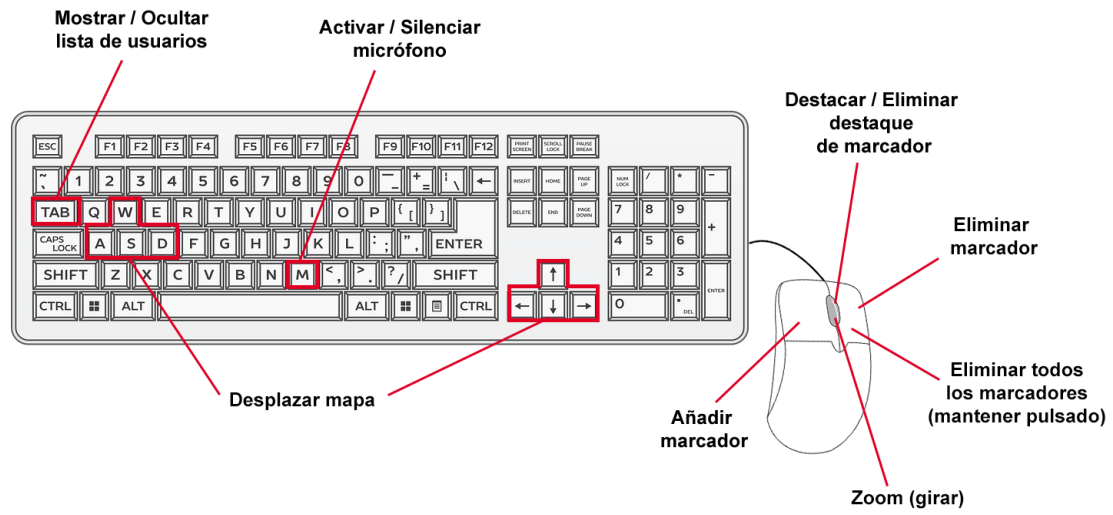


Fig. 6.6. Controles de juego en entorno sin realidad virtual

En cuanto al entorno con realidad virtual, se utilizan los controles Oculus Touch, los cuales proporcionan una experiencia más inmersiva. Los controles asociados al uso de marcadores se asignan al control derecho, mientras que la interacción con el mapa se realiza con el control izquierdo. De esta manera, se intenta facilitar el aprendizaje por parte del usuario. Los controles de juego se muestran a través de la tabla 6.19 y la figura 6.7:

TABLA 6.19: CONTROLES DE JUEGO EN ENTORNO CON REALIDAD VIRTUAL

<b>Función</b>	<b>Controles de Oculus Touch</b>
Desplazar mapa hacia arriba	Mover joystick hacia delante
Desplazar mapa hacia abajo	Mover joystick hacia atrás
Desplazar mapa hacia la derecha	Mover joystick hacia la derecha
Desplazar mapa hacia la izquierda	Mover joystick hacia la izquierda
<i>Zoom in</i>	Gatillo de índice izquierdo
<i>Zoom out</i>	Gatillo de mano izquierdo
Añadir marcador	Gatillo de índice derecho
Eliminar marcador	Gatillo de mano derecho
Eliminar todos los marcadores propios	Mantener pulsado gatillo de mano derecho durante un segundo
Destacar marcador	A sobre un marcador que no esté destacado por el usuario
Eliminar destaque de marcador	A sobre un marcador que sí esté destacado por el usuario
Mostrar / ocultar lista de usuarios	X
Activar / silenciar micrófono	Y
Activar / desactivar rayo	Pulsar joystick derecho
Salir de la aplicación	ESC (teclado)

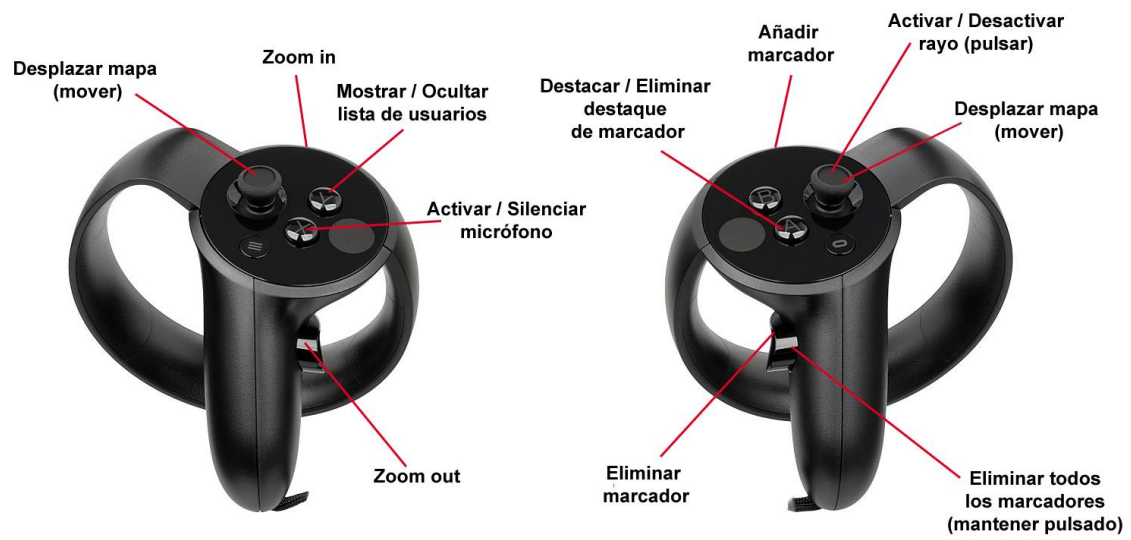


Fig. 6.7. Controles de juego en entorno con realidad virtual

## 7. IMPLEMENTACIÓN

En este capítulo, se tratan los aspectos más importantes de la implementación de la aplicación, sobre todo en lo relativo a la colaboración. Se detalla el planteamiento para resolver cada una de las necesidades que resuelve la aplicación. Además, se tratarán los problemas encontrados a lo largo de la implementación, así como las decisiones que se han tomado al respecto. Este apartado se divide en varias secciones, de acuerdo con las distintas funcionalidades implementadas.

### 7.1. Escenas

Se han creado dos escenas diferentes. La primera se trata de una escena de configuración. Esta contiene un *canvas* (área donde se incluyen elementos de interfaz de usuario), en el cual se introduce el alias (nombre de usuario) y se selecciona el tipo de entorno (con o sin realidad virtual). Al pulsar el botón *Start*, se activarán los controles asociados a la configuración elegida y se cargará la segunda escena. Esta es la escena principal, la cual contiene el entorno de interacción y visualización del mapa. En la figura 7.1 se muestra una instantánea de cada una de las escenas descritas.

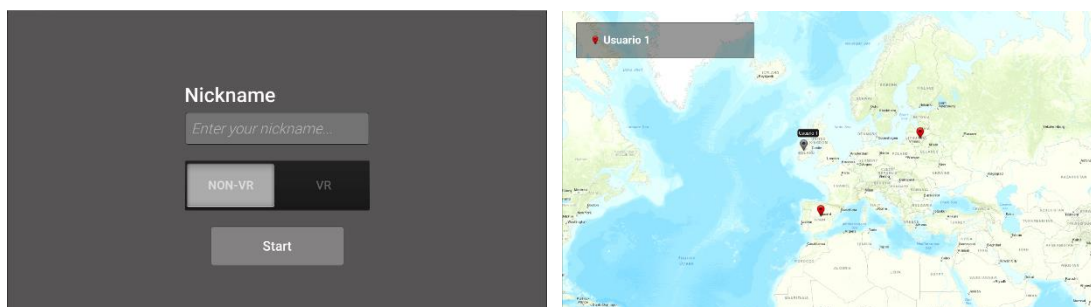


Fig. 7.1. Escena inicial de configuración (izquierda) y escena principal (derecha)

Se crearon dos cámaras diferentes que enfocan al mapa, una para cada tipo de entorno. Esto se debe a que el campo de visión en realidad virtual es mayor, lo cual implica que, para observar el mismo contenido del mapa al iniciar la aplicación, la cámara del entorno sin VR debía ser más cercana.

La cámara sin VR se posicionó de tal forma que, en el nivel de *zoom* mínimo, se observa el mapa completo. Así, el entorno sin VR es bidimensional, ya que el mapa invade toda la pantalla. No obstante, esta transformación de un entorno 3D a uno en 2D provocó un problema: los marcadores no se añadían en la posición en la que se colocaba el cursor, sino que existía un ligero desfase, el cual aumentaba al alejarse del centro del mapa.

El problema se solucionó cambiando la proyección de la cámara de perspectiva a ortográfica. De esta forma, no se tenía en cuenta la tercera dimensión, y la posición indicada correspondía con la proyección ortogonal del cursor sobre el mapa.

## 7.2. Interacción con mapas y uso de marcadores

En este caso, la librería *OnlineMaps* ofrece un conjunto de operaciones básicas para el tratamiento de mapas. Tal y como se definió en el capítulo 6, dedicado al diseño del sistema, se creará un componente llamado controlador de juego (*GameController*), y otro componente denominado controlador de mapas (*MapController*). El primero interpretará las acciones del usuario y, posteriormente, llamará a los métodos correspondientes del controlador de mapas. Este último utilizará las funciones de *OnlineMaps* para realizar las modificaciones correspondientes sobre el mapa.

La navegación a través del mapa, es decir, el desplazamiento y el nivel de *zoom* de este, se basará en la modificación de dos atributos que ofrece el mapa: las coordenadas de su centro, expresadas en coordenadas geográficas, y un número entero que indica el nivel de *zoom*.

Es necesario tener en cuenta que, a mayor nivel de *zoom*, menor debe ser la distancia de desplazamiento. Por ello, se define una distancia de desplazamiento diferente para cada nivel de *zoom*. Asimismo, al tratarse de coordenadas geográficas y no cartesianas, tampoco existirá coincidencia entre el desplazamiento vertical y horizontal.

En lo relativo al uso de marcadores, se implementó la creación, eliminación, destaque y eliminación de destaque de los marcadores. Además, se implementaron las rutinas que controlan la textura de los marcadores que son señalados.

La librería *OnlineMaps* proporciona una clase llamada *OnlineMapsMarker*, la cual representa cada uno de los marcadores que son añadidos al mapa. Esta clase ha sido modificada para almacenar información útil para la colaboración, como el identificador de su propietario, su color o su color de destaque. Para la utilización del rayo, así como el cálculo de la incidencia de este sobre el mapa, el controlador de mapas se apoya en la librería *VRTK*. Con el uso de estas dos clases, se ha implementado lo siguiente:

- **Añadir marcadores.** se indica la posición del mapa en la que se quiere añadir y su textura correspondiente, la cual depende del color asociado del usuario. Para ello, se utiliza un array de texturas, en el que cada posición almacena un marcador de cada color. Cuando un marcador es añadido, se incluye el identificador del usuario propietario.
- **Señalar marcadores.** En cada fotograma, se comprueba la posición del mapa que está señalando con el método de entrada correspondiente (cursor del ratón, retículo o rayo). Después, se obtiene el objeto asociado al marcador que está siendo señalado. De esta forma, es posible cambiar su color a gris al señalarlo, así como devolverle su textura original cuando este deja de ser señalado. Por esta razón, se ha añadido a la clase *OnlineMapsMarker* el identificador de su color y su color de destaque actual como nuevos atributos. Este último tendrá valor igual a -1 cuando no se encuentre destacado.
- **Eliminar marcadores.** En el caso de la eliminación de un marcador, se indica a *OnlineMaps* el marcador que se está señalando cuando se realiza la acción, y este es eliminado del mapa. Esto se realizará solo si el usuario ha creado ese marcador, es decir, el identificador del propietario del marcador coincide con el del usuario que intenta eliminarlo.

- **Eliminar todos los marcadores propios.** No es necesario señalar ningún marcador. Consiste en la eliminación de todos los marcadores con el identificador de propietario concreto.
- **Destacar marcadores.** Cuando un marcador es destacado, se modifica su tamaño y su textura. La textura es modificada añadiendo un círculo que rodea el marcador. Si se elimina el destaque, se devuelve al marcador su tamaño y textura iniciales.

### 7.3. Colaboración

Una vez se ha implementado la interacción individual de cada usuario con su entorno propio, se procede a desarrollar la colaboración entre ellos. Para conseguir tal fin, se hace uso de la librería *Photon Unity Networking*.

En primer lugar, se enumeran los elementos principales que se han utilizado para conseguir la colaboración:

- **Clase *PhotonView*.** Este script debe asociarse a cualquier objeto que se quiera identificar a través de la red. En este caso, el objeto representará cada usuario, el cual debe instanciarse en la sala. Por ejemplo, si hay cuatro usuarios en la misma sala, la aplicación contendrá cuatro objetos que representan al usuario.
- **Clase *Room*.** Se trata de la sala en la que se encuentra el usuario. En este caso, existe una única sala.
- **Clase *PhotonPlayer*.** Representa a un jugador o usuario, y contiene toda su información. Posee un identificador único dentro de la sala.
- ***RaiseEvent*.** Es un tipo de llamada a procedimientos remotos que ofrece *Photon*. Consiste en que un usuario llame a una función, y esta se ejecute en el resto de los usuarios (o aquellos que este elija). En otras palabras, es una forma de enviar información a través de la red y tratar eventos personalizados. El evento se identifica con un código de operación, y se recibe el identificador del usuario que ha realizado la llamada.
- **Funciones extendidas de *Photon*.** Se trata de funciones que se ejecutan cuando ocurre un evento relacionado con el estado de conexión de un usuario en la red. Se crea un gestor de conexiones, que contendrá el código de tratamiento de este tipo de eventos. Por ejemplo, el método *OnJoinedRoom()* se ejecuta cuando el usuario se une a una sala, y *OnPhotonPlayerConnected (PhotonPlayer newPlayer)* cuando otro usuario se haya conectado.

Las clases *PhotonPlayer* y *Room* ofrecen la posibilidad de almacenar propiedades personalizadas (*CustomProperties*), a las que podrá acceder cualquier usuario que se encuentre en la sala. Esta información compartida será útil para lograr la coherencia y consistencia de los mapas de todos los usuarios.

Tras conocer los elementos principales de *Photon Unity Networking*, se desarrollan las soluciones implementadas sobre cada funcionalidad para conseguir la colaboración:

- **Elección de color.** No pueden existir dos usuarios que creen marcadores del mismo color, por lo que debe existir un algoritmo que controle este aspecto. Se ha creado una propiedad personalizada de la sala, la cual es un array de ocho números enteros. La posición del array representará uno de los ocho colores de marcadores, y el contenido de cada posición será el identificador del usuario que utiliza el color. Este valor será -1 si ningún usuario está utilizando ese color.

Cuando un usuario entre en la sala, recorrerá el array hasta que encuentre un valor igual a -1 o un identificador de un usuario que ya no se encuentre en la sala. En ese caso el color es escogido y la propiedad de sala es actualizada.

- **Añadir marcadores.** Cuando un usuario añade un marcador, ejecuta un *RaiseEvent*, en el cual indica la posición en la que este ha sido colocado. A partir del identificador del usuario que ha realizado la llamada, se obtiene su color asociado, el cual es una de las propiedades personalizadas de los usuarios. De esta forma, el resto de los usuarios pueden añadir el marcador del mismo color en sus respectivos mapas.
- **Eliminar marcadores.** En este caso, también se envía la posición del marcador. El resto de los usuarios obtienen la lista de todos los marcadores del mapa (a través de *OnlineMaps*) y eliminan el marcador cuya posición coincide con la recibida.
- **Eliminar todos los marcadores de un usuario.** Se trata de un evento independiente al de eliminar un único marcador. Se recorre la lista de marcadores, y se eliminan todos aquellos en los que el identificador de propietario coincida con el identificador del llamante.
- **Destacar marcadores.** La resolución es similar a la eliminación de un marcador. El parámetro enviado por el llamante es la posición del marcador destacado. El resto de los usuarios recorren la lista de marcadores y modifican la textura de este. Se utilizará un código de operación distinto para la eliminación del destaque del marcador, pero el procedimiento es similar, con la diferencia de que la textura final es diferente.
- **Flechas que apuntan marcadores destacados.** Una posible implementación consiste en recorrer la lista de marcadores cada fotograma y, en caso de que el marcador esté destacado (color de destaque no es igual a -1), se calcula si se encuentra fuera de los límites del mapa (no es visible por el usuario). En caso afirmativo, se muestra la flecha que lo señala. Se trata de una solución que puede generar problemas de rendimiento cuando el número de marcadores es elevado.

Para evitar tener que recorrer la lista de todos los marcadores en cada fotograma, se ha creado una segunda propiedad personalizada de sala. Se trata de un array de ocho posiciones de tipo *Vector2*, en el que cada posición identifica a un color, y contiene las coordenadas geográficas del marcador que está destacado con dicho color. Si el marcador no se encuentra destacado, su valor



es (*int.MinValue*, *int.MaxValue*). Esta información debe actualizarse cada vez que se destaca o se elimina el destaque de un marcador.

Con la propiedad personalizada creada, solo es necesario recorrer un array de ocho posiciones para mostrar las flechas correspondientes. La complejidad computacional reside en el cálculo de la posición y la rotación de cada una de las flechas, así como saber cuándo el marcador se encuentra fuera del plano del mapa.

La rotación de la flecha se obtiene calculando el ángulo  $\alpha$  entre la componente horizontal y la recta que pasa por el origen y el marcador correspondiente (denotada como  $r5$ ). En cuanto a la posición de la flecha, se crea un rectángulo concéntrico al mapa, el cual representará todos aquellos puntos en los que se puede colocar el marcador. Después, se calcula el punto de corte entre  $r5$  y el rectángulo creado. Dependiendo del valor del ángulo  $\alpha$ , se calculará la intersección con una de las cuatro rectas que forman el rectángulo ( $r1$ ,  $r2$ ,  $r3$  o  $r4$ ). Para obtener la posición de la flecha, se resuelve un sistema de ecuaciones lineales con dos incógnitas. En la figura 7.2 se muestra de forma gráfica el planteamiento que se acaba de describir.

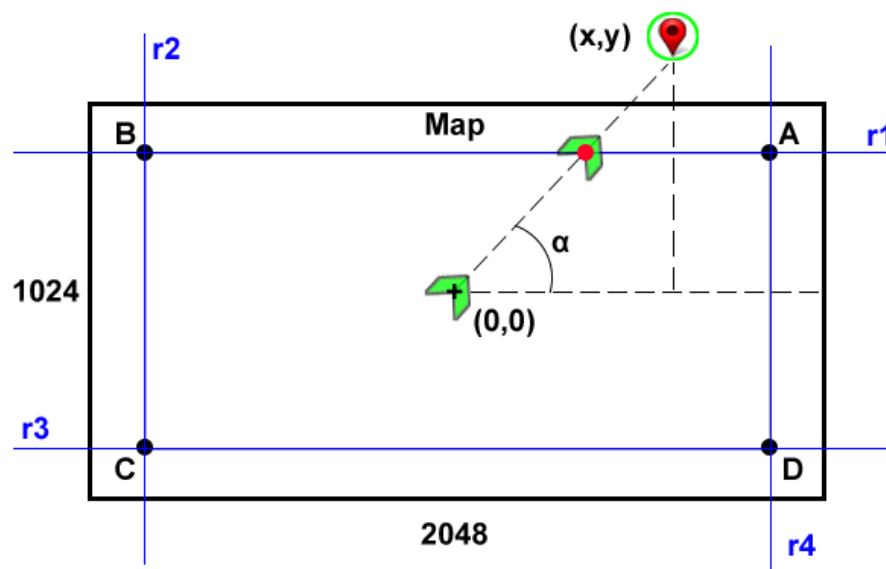


Fig. 7.2. Cálculo de la rotación y la posición de las flechas

Además, se ha creado un script encargado de activar o desactivar las flechas, en función de la existencia de marcadores destacados fuera del plano del mapa. Para ello, se calculan las coordenadas geográficas de las cuatro esquinas del mapa, y se comprueba con el array de coordenadas de marcadores destacados si algún marcador se encuentra fuera de los límites.

Cabe destacar que, para la realización de los cálculos, en el caso del entorno sin realidad virtual, en lugar de tener en cuenta el mapa completo, se opera sobre la parte visible del mapa. Esto puede ocurrir si la relación de aspecto de la pantalla no es 2:1.

- **Chat de voz.** La implementación del chat de voz ha sido sencilla debido a la utilización de la librería *PhotonVoice*. Se asociaron dos scripts al objeto del usuario que será instanciado: *PhotonVoiceRecorder* y *PhotonVoiceSpeaker*. El primero se encarga de transmitir los datos de voz, mientras que el segundo los reproduce.

Si el objeto es el usuario local, únicamente se ejecutará *PhotonVoiceRecorder*, ya que un usuario transmite voz y no debe escucharse a sí mismo. En el caso de los usuarios remotos, se ejecutará *PhotonVoiceSpeaker*. Se desactivará la transmisión de audio para silenciar el micrófono.

Además, se han creado dos propiedades de usuario que serán útiles para el mostrado de la lista de usuarios. Las dos serán variables booleanas, e indicarán si un usuario está transmitiendo y si el micrófono se encuentra silenciado o no.

- **Lista de usuarios.** La clase *PhotonPlayer* almacena todos los datos necesarios para el mostrado de la lista de usuarios. Se obtiene la lista de usuarios que se encuentran en la sala y se muestra la información correspondiente.
- **Añadir marcadores al entrar en sala.** Cuando un usuario se conecta, el resto de los usuarios ejecutan la función extendida *OnPhotonPlayerConnected* (*PhotonPlayer newPlayer*). En ella, ejecutan un *RaiseEvent*, mediante el cual cada uno envía al usuario recién conectado una lista con las coordenadas de todos los marcadores que ha añadido. El usuario trata cada evento añadiendo los marcadores en las coordenadas recibidas.
- **Eliminar marcadores cuando un usuario sale de la sala.** Cuando un usuario se desconecta, automáticamente los usuarios ejecutan la función *OnPhotonPlayerDisconnected* (*PhotonPlayer otherPlayer*), en la cual eliminan todos los marcadores cuyo propietario sea el usuario que se acaba de desconectar.

## 8. PRUEBAS

Después de realizar la implementación de la aplicación, es necesario comprobar que esta se ajusta a las necesidades descritas a lo largo de las diferentes fases del proyecto. Para ello, se especificará una batería de pruebas que verifique el correcto funcionamiento de la aplicación. A continuación, se realizarán las pruebas especificadas, anotando los resultados obtenidos. De esta forma, se recogerán los errores cometidos. Esta información será útil para mejorar la aplicación en sus posibles versiones posteriores.

### 8.1. Especificación del conjunto de pruebas

En este apartado, se definirá una batería de pruebas con la que se verificará si el sistema funciona correctamente. Estas pruebas estarán enfocadas a comprobar que la aplicación abarca todas las funcionalidades descritas en el capítulo dedicado al análisis. Se trata de pruebas principalmente funcionales, a través de las cuales se constatará que todas las capacidades definidas en la especificación de requisitos de software han sido cubiertas. Debido a que la aplicación se puede ejecutar en dos entornos distintos (con y sin realidad virtual), existirán pruebas que comprueben la misma funcionalidad, pero en entornos diferentes.

Cabe destacar que, posteriormente a este proyecto, se realizarán experimentos en los que los usuarios potenciales de la aplicación probarán la aplicación desarrollada. De esta forma, podrá comprobarse la utilidad del sistema para la visualización e interacción con mapas, así como el grado de usabilidad del este. Además, se recogerán los errores cometidos y las posibles mejoras que pueden realizarse.

La especificación de los casos de prueba se realizará a través del uso de tablas. Estas seguirán la plantilla mostrada en la tabla 8.1, la cual contiene las siguientes propiedades.

- **Identificador.** Cada caso de prueba posee un identificador único. La nomenclatura utilizada es del formato CP-XX, donde CP indica que se trata de un caso de prueba, y XX representa un número entero de dos cifras que comienza por el número uno (01) y se va incrementando de unidad en unidad.
- **Objetivo.** Breve descripción de la funcionalidad que se comprueba.
- **Entorno.** Es el entorno en el que se da la prueba. Puede ser realidad virtual, sin realidad virtual o ambos. También se indica el número de usuarios necesarios.
- **Acciones.** Descripción de los pasos que se deben realizar para llevar a cabo la prueba.
- **Resultado esperado.** Respuesta esperada del sistema ante la prueba realizada.
- **Componentes.** Identificadores de los componentes implicados en la respuesta del sistema.
- **Requisitos.** Identificadores de los principales requisitos de software que verifica el caso de prueba. En el caso de que sean requisitos de información, se incluirá cuando el éxito de la prueba depende de que ese dato sea correcto.

TABLA 8.1: PLANTILLA DE CASOS DE PRUEBA

<b>CP-XX</b>	
<b>Objetivo</b>	
<b>Entorno</b>	
<b>Acciones</b>	
<b>Resultado esperado</b>	
<b>Componentes</b>	
<b>Requisitos</b>	

Las primeras pruebas se basarán en comprobar la interacción de forma individual con el mapa, incluyendo el uso de marcadores. Posteriormente, las pruebas verificarán si la colaboración se realiza correctamente.

A continuación, se describe el conjunto de pruebas que se van a realizar:

TABLA 8.2: CASO DE PRUEBA CP-01

<b>CP-01</b>	
<b>Objetivo</b>	Verificar inicio y entorno
<b>Entorno</b>	Sin realidad virtual, un usuario
<b>Acciones</b>	Iniciar la aplicación eligiendo entorno sin realidad virtual.
<b>Resultado esperado</b>	Se muestra el mapa en la pantalla.
<b>Componentes</b>	CO-03, CO.10, CO-12
<b>Requisitos</b>	RS-F01, RS-F44

TABLA 8.3: CASO DE PRUEBA CP-02

<b>CP-02</b>	
<b>Objetivo</b>	Verificar el funcionamiento del zoom del mapa
<b>Entorno</b>	Sin realidad virtual, un usuario
<b>Acciones</b>	<p>Iniciar la aplicación eligiendo entorno sin realidad virtual.</p> <p>Colocar el cursor en la posición donde se quiere ampliar o alejar el mapa.</p> <p>Ampliar el mapa girando la rueda del ratón hacia delante (<i>scroll up</i>) y alejarlo girando la misma rueda hacia atrás (<i>scroll down</i>).</p>
<b>Resultado esperado</b>	El mapa posee varios niveles de <i>zoom</i> , y el centro de este pasa a ser el lugar donde estaba situado el cursor en el momento de realizar cada acción. Existe límite superior e inferior de ampliación del mapa, y la aplicación devuelve un sonido de error cuando estos se alcanzan.
<b>Componentes</b>	CO-01, CO-03, CO-06, CO-07
<b>Requisitos</b>	RS-F02, RS-F03, RS-F04, RS-F05, RS-F06, RS-F07

TABLA 8.4: CASO DE PRUEBA CP-03

<b>CP-03</b>	
<b>Objetivo</b>	Verificar el desplazamiento del mapa
<b>Entorno</b>	Sin realidad virtual, un usuario
<b>Acciones</b>	<p>Iniciar la aplicación eligiendo entorno sin realidad virtual.</p> <p>Desplazar el mapa pulsando las flechas de dirección del teclado.</p>
<b>Resultado esperado</b>	El mapa se desplaza en la dirección indicada.
<b>Componentes</b>	CO-01, CO-03, CO-06, CO-07
<b>Requisitos</b>	RS-F08

TABLA 8.5: CASO DE PRUEBA CP-04

<b>CP-04</b>	
<b>Objetivo</b>	Verificar la adición de marcadores
<b>Entorno</b>	Sin realidad virtual, un usuario
<b>Acciones</b>	Iniciar la aplicación eligiendo entorno sin realidad virtual. Añadir marcadores colocando el cursor en la posición en la que se quiere añadir el marcador y pulsando el botón izquierdo del ratón.
<b>Resultado esperado</b>	Los marcadores se añaden en las posiciones en las que se encontraba el cursor en el momento de añadirlos.
<b>Componentes</b>	CO-01, CO-03, CO-06, CO-07, CO-11, CO-12
<b>Requisitos</b>	RS-F10

TABLA 8.6: CASO DE PRUEBA CP-05

<b>CP-05</b>	
<b>Objetivo</b>	Verificar la eliminación de marcadores
<b>Entorno</b>	Sin realidad virtual, un usuario
<b>Acciones</b>	Iniciar la aplicación eligiendo entorno sin realidad virtual. Añadir varios marcadores. Colocar el cursor encima de cada marcador que se quiere eliminar y pulsar el botón derecho del ratón para eliminarlo.
<b>Resultado esperado</b>	Se eliminan del mapa los marcadores deseados.
<b>Componentes</b>	CO-01, CO-03, CO-06, CO-07, CO-11, CO-12
<b>Requisitos</b>	RS-F14

TABLA 8.7: CASO DE PRUEBA CP-06

<b>CP-06</b>	
<b>Objetivo</b>	Verificar el cambio de color del marcador señalado
<b>Entorno</b>	Sin realidad virtual, un usuario
<b>Acciones</b>	<p>Iniciar la aplicación eligiendo entorno sin realidad virtual.</p> <p>Añadir varios marcadores.</p> <p>Colocar el cursor encima los marcadores creados.</p>
<b>Resultado esperado</b>	Cada vez que se coloca el cursor encima de un marcador, su color pasa a ser gris. Cuando el cursor deja de estar encima del marcador, este vuelve a su color original.
<b>Componentes</b>	CO-03, CO-06
<b>Requisitos</b>	RS-F16

TABLA 8.8: CASO DE PRUEBA CP-07

<b>CP-07</b>	
<b>Objetivo</b>	Verificar el destaque de marcadores
<b>Entorno</b>	Sin realidad virtual, un usuario
<b>Acciones</b>	<p>Iniciar la aplicación eligiendo entorno sin realidad virtual.</p> <p>Añadir varios marcadores.</p> <p>Colocar el cursor encima de cada marcador que se quiere destacar y pulsar el botón central del ratón para que se destaquen.</p>
<b>Resultado esperado</b>	Cada vez que se destaca un marcador, su tamaño aumenta y se rodea con un círculo del mismo color. Además, solo puede estar destacado un marcador a la vez.
<b>Componentes</b>	CO-01, CO-03, CO-06, CO-07, CO-11, CO-12
<b>Requisitos</b>	RS-F18, RS-F19

TABLA 8.9: CASO DE PRUEBA CP-08

<b>CP-08</b>	
<b>Objetivo</b>	Verificar la eliminación de destaque de un marcador
<b>Entorno</b>	Sin realidad virtual, un usuario
<b>Acciones</b>	<p>Iniciar la aplicación eligiendo entorno sin realidad virtual.</p> <p>Añadir varios marcadores. Destacar un marcador.</p> <p>Señalando el marcador con el cursor, pulsar el botón central del ratón para deshacer el destaque.</p>
<b>Resultado esperado</b>	El marcador que se había destacado ha recuperado su tamaño inicial y desaparece el círculo que lo rodeaba. No existe ningún marcador que destaque sobre el resto.
<b>Componentes</b>	CO-01, CO-03, CO-06, CO-07, CO-11, CO-12
<b>Requisitos</b>	RS-F24

TABLA 8.10: CASO DE PRUEBA CP-09

<b>CP-09</b>	
<b>Objetivo</b>	Verificar la eliminación de todos los marcadores propios
<b>Entorno</b>	Sin realidad virtual, un usuario
<b>Acciones</b>	<p>Iniciar la aplicación eligiendo entorno sin realidad virtual.</p> <p>Añadir varios marcadores.</p> <p>Mantener pulsado el botón derecho del ratón durante un segundo para eliminar todos los marcadores.</p>
<b>Resultado esperado</b>	Se eliminan todos los marcadores del mapa.
<b>Componentes</b>	CO-01, CO-03, CO-06, CO-07, CO-11, CO-12
<b>Requisitos</b>	RS-F27



TABLA 8.11: CASO DE PRUEBA CP-10

<b>CP-10</b>	
<b>Objetivo</b>	Verificar las flechas que señalan marcadores destacados
<b>Entorno</b>	Sin realidad virtual, un usuario
<b>Acciones</b>	<p>Iniciar la aplicación eligiendo entorno sin realidad virtual.</p> <p>Añadir un marcador.</p> <p>Destacar el marcador añadido.</p> <p>Desplazar el mapa, de tal forma que el marcador destacado desaparezca de la pantalla.</p>
<b>Resultado esperado</b>	Mientras el marcador destacado es visible, no se muestran las flechas. Cuando el marcador deja de ser visible, las flechas se muestran señalando la posición de este. El color de la flecha coincide con el del círculo que rodea el marcador señalado.
<b>Componentes</b>	CO-01, CO-03, CO-04, CO-08
<b>Requisitos</b>	RS-F37

TABLA 8.12: CASO DE PRUEBA CP-11

<b>CP-11</b>	
<b>Objetivo</b>	Verificar inicio y entorno
<b>Entorno</b>	Realidad virtual, un usuario
<b>Acciones</b>	Iniciar la aplicación eligiendo entorno con realidad virtual.
<b>Resultado esperado</b>	Se puede observar el entorno de realidad virtual, en el cual se sitúa un plano vertical que contiene el mapa.
<b>Componentes</b>	CO-03, CO.10, CO-12
<b>Requisitos</b>	RS-F01, RS-F44

TABLA 8.13: CASO DE PRUEBA CP-12

<b>CP-12</b>	
<b>Objetivo</b>	Verificar el funcionamiento del zoom del mapa
<b>Entorno</b>	Realidad virtual, un usuario
<b>Acciones</b>	<p>Iniciar la aplicación eligiendo entorno con realidad virtual.</p> <p>Colocar el retículo (centro de la mirada) en la posición donde se quiere ampliar o alejar el mapa.</p> <p>Ampliar el mapa pulsando el gatillo de índice izquierdo de Oculus Touch y alejarlo pulsando el gatillo de mano izquierdo de Oculus Touch.</p>
<b>Resultado esperado</b>	El mapa posee varios niveles de <i>zoom</i> , y el centro de este pasa a ser el lugar donde estaba situado el retículo en el momento de realizar cada acción. Existe límite superior e inferior de ampliación del mapa, y la aplicación devuelve un sonido de error cuando estos se alcanzan.
<b>Componentes</b>	CO-01, CO-03, CO-06, CO-07
<b>Requisitos</b>	RS-F02, RS-F03, RS-F04, RS-F05, RS-F06, RS-F04

TABLA 8.14: CASO DE PRUEBA CP-13

<b>CP-13</b>	
<b>Objetivo</b>	Verificar el desplazamiento del mapa
<b>Entorno</b>	Realidad virtual, un usuario
<b>Acciones</b>	<p>Iniciar la aplicación eligiendo entorno con realidad virtual.</p> <p>Desplazar el mapa moviendo el joystick derecho y/o izquierdo de Oculus Touch.</p>
<b>Resultado esperado</b>	El mapa se desplaza en la dirección indicada.
<b>Componentes</b>	CO-01, CO-03, CO-06, CO-07
<b>Requisitos</b>	RS-F08

TABLA 8.15: CASO DE PRUEBA CP-14

<b>CP-14</b>	
<b>Objetivo</b>	Verificar la adición de marcadores mediante retículo
<b>Entorno</b>	Realidad virtual, un usuario
<b>Acciones</b>	<p>Iniciar la aplicación eligiendo entorno con realidad virtual.</p> <p>Añadir marcadores colocando el retículo (centro de la mirada) en la posición en la que se quiere añadir el marcador y pulsando el gatillo de índice derecho de Oculus Touch.</p>
<b>Resultado esperado</b>	Los marcadores se añaden en las posiciones en las que se encontraba el retículo en el momento de añadirlos.
<b>Componentes</b>	CO-01, CO-03, CO-06, CO-07, CO-11, CO-12
<b>Requisitos</b>	RS-F10, RS-F17

TABLA 8.16: CASO DE PRUEBA CP-15

<b>CP-15</b>	
<b>Objetivo</b>	Verificar la adición de marcadores mediante rayo
<b>Entorno</b>	Realidad virtual, un usuario
<b>Acciones</b>	<p>Iniciar la aplicación eligiendo entorno con realidad virtual.</p> <p>Activar rayo pulsando el joystick derecho de Oculus Touch.</p> <p>Añadir marcadores apuntando con el rayo la posición en la que se quiere añadir el marcador y pulsando el gatillo de índice derecho de Oculus Touch.</p>
<b>Resultado esperado</b>	Los marcadores se añaden en las posiciones en las que apuntaba el rayo en el momento de añadirlos.
<b>Componentes</b>	CO-01, CO-02, CO-03, CO-06, CO-07, CO-11, CO-12
<b>Requisitos</b>	RS-F10, RS-F17

TABLA 8.17: CASO DE PRUEBA CP-16

<b>CP-16</b>	
<b>Objetivo</b>	Verificar la eliminación de marcadores mediante retículo
<b>Entorno</b>	Realidad virtual, un usuario
<b>Acciones</b>	<p>Iniciar la aplicación eligiendo entorno con realidad virtual.</p> <p>Añadir varios marcadores.</p> <p>Colocar el retículo encima de cada marcador que se quiere eliminar y accionar el gatillo de mano derecho de Oculus Touch para eliminarlo.</p>
<b>Resultado esperado</b>	Se eliminan del mapa los marcadores deseados.
<b>Componentes</b>	CO-01, CO-03, CO-06, CO-07, CO-11, CO-12
<b>Requisitos</b>	RS-F14, RS-F17

TABLA 8.18: CASO DE PRUEBA CP-17

<b>CP-17</b>	
<b>Objetivo</b>	Verificar la eliminación de marcadores mediante rayo
<b>Entorno</b>	Realidad virtual, un usuario
<b>Acciones</b>	<p>Iniciar la aplicación eligiendo entorno con realidad virtual.</p> <p>Activar rayo pulsando el joystick derecho de Oculus Touch.</p> <p>Apuntar con el rayo al marcador que se quiere eliminar y accionar el gatillo de mano derecho de Oculus Touch para eliminarlo.</p>
<b>Resultado esperado</b>	Se eliminan del mapa los marcadores deseados.
<b>Componentes</b>	CO-01, CO-02, CO-03, CO-06, CO-07, CO-11, CO-12
<b>Requisitos</b>	RS-F14, RS-F17

TABLA 8.19: CASO DE PRUEBA CP-18

<b>CP-18</b>	
<b>Objetivo</b>	Verificar el cambio de color del marcador señalado con retículo
<b>Entorno</b>	Realidad virtual, un usuario
<b>Acciones</b>	<p>Iniciar la aplicación eligiendo entorno con realidad virtual.</p> <p>Añadir varios marcadores.</p> <p>Colocar el retículo encima los marcadores creados.</p>
<b>Resultado esperado</b>	Cada vez que se coloca el retículo encima de un marcador, su color pasa a ser gris. Cuando el retículo deja de estar encima del marcador, este vuelve a su color original.
<b>Componentes</b>	CO-03, CO-06
<b>Requisitos</b>	RS-F16, RS-F17

TABLA 8.20: CASO DE PRUEBA CP-19

<b>CP-19</b>	
<b>Objetivo</b>	Verificar el cambio de color del marcador señalado con rayo
<b>Entorno</b>	Realidad virtual, un usuario
<b>Acciones</b>	<p>Iniciar la aplicación eligiendo entorno con realidad virtual.</p> <p>Activar rayo pulsando el joystick derecho de Oculus Touch.</p> <p>Añadir varios marcadores.</p> <p>Apuntar con el rayo a los marcadores creados.</p>
<b>Resultado esperado</b>	Cada vez que se apunta con el rayo a un marcador, su color pasa a ser gris. Cuando el cursor deja apuntar el marcador, este vuelve a su color original.
<b>Componentes</b>	CO-02, CO-03, CO-06
<b>Requisitos</b>	RS-F16, RS-F17

TABLA 8.21: CASO DE PRUEBA CP-20

<b>CP-20</b>	
<b>Objetivo</b>	Verificar el destaque de marcadores mediante retículo
<b>Entorno</b>	Realidad virtual, un usuario
<b>Acciones</b>	<p>Iniciar la aplicación eligiendo entorno con realidad virtual.</p> <p>Añadir varios marcadores.</p> <p>Colocar el retículo encima de cada marcador que se quiere destacar y pulsar el botón 'A' de Oculus Touch.</p>
<b>Resultado esperado</b>	<p>Cada vez que se destaca un marcador, su tamaño aumenta y se rodea con un círculo del mismo color.</p> <p>Además, solo puede estar destacado un marcador a la vez.</p>
<b>Componentes</b>	CO-01, CO-03, CO-06, CO-07, CO-11, CO-12
<b>Requisitos</b>	RS-F18, RS-F19, RS-F17

TABLA 8.22: CASO DE PRUEBA CP-21

<b>CP-21</b>	
<b>Objetivo</b>	Verificar el destaque de marcadores mediante rayo
<b>Entorno</b>	Realidad virtual, un usuario
<b>Acciones</b>	<p>Iniciar la aplicación eligiendo entorno con realidad virtual.</p> <p>Activar rayo pulsando el joystick derecho de Oculus Touch.</p> <p>Añadir varios marcadores.</p> <p>Apuntar con el rayo a cada marcador que se quiere destacar y pulsar el botón 'A' de Oculus Touch.</p>
<b>Resultado esperado</b>	<p>Cada vez que se destaca un marcador, su tamaño aumenta y se rodea con un círculo del mismo color.</p> <p>Además, solo puede estar destacado un marcador a la vez.</p>
<b>Componentes</b>	CO-01, CO-02, CO-03, CO-06, CO-07, CO-11, CO-12
<b>Requisitos</b>	RS-F18, RS-F19, RS-F17

TABLA 8.23: CASO DE PRUEBA CP-22

<b>CP-22</b>	
<b>Objetivo</b>	Verificar la eliminación de destaque de un marcador mediante retículo
<b>Entorno</b>	Realidad virtual, un usuario
<b>Acciones</b>	<p>Iniciar la aplicación eligiendo entorno con realidad virtual.</p> <p>Añadir varios marcadores. Destacar un marcador.</p> <p>Mientras se señala el marcador con el retículo, pulsar el botón 'A' de Oculus Touch para deshacer el destaque.</p>
<b>Resultado esperado</b>	El marcador que se había destacado ha recuperado su tamaño inicial y desaparece el círculo que lo rodeaba. No existe ningún marcador que destaque sobre el resto.
<b>Componentes</b>	CO-01, CO-03, CO-06, CO-07, CO-11, CO-12
<b>Requisitos</b>	RS-F24, RS-F17

TABLA 8.24: CASO DE PRUEBA CP-23

<b>CP-23</b>	
<b>Objetivo</b>	Verificar la eliminación de destaque de un marcador mediante rayo
<b>Entorno</b>	Realidad virtual, un usuario
<b>Acciones</b>	<p>Iniciar la aplicación eligiendo entorno con realidad virtual.</p> <p>Activar rayo pulsando el joystick derecho de Oculus Touch.</p> <p>Añadir varios marcadores. Destacar un marcador.</p> <p>Mientras se apunta al marcador con el rayo, pulsar el botón 'A' de Oculus Touch para deshacer el destaque.</p>
<b>Resultado esperado</b>	El marcador que se había destacado ha recuperado su tamaño inicial y desaparece el círculo que lo rodeaba. No existe ningún marcador que destaque sobre el resto.
<b>Componentes</b>	CO-01, CO-02, CO-03, CO-06, CO-07, CO-11, CO-12
<b>Requisitos</b>	RS-F24, RS-F17

TABLA 8.25: CASO DE PRUEBA CP-24

<b>CP-24</b>	
<b>Objetivo</b>	Verificar la no simultaneidad de entrada de retículo y rayo
<b>Entorno</b>	Realidad virtual, un usuario
<b>Acciones</b>	<p>Iniciar la aplicación eligiendo entorno con realidad virtual.</p> <p>Añadir, eliminar y destacar marcadores.</p> <p>Activar y desactivar rayo pulsando el joystick derecho de Oculus Touch. Mantener activado el rayo.</p> <p>Añadir, eliminar y destacar marcadores.</p>
<b>Resultado esperado</b>	<p>El rayo puede activarse y desactivarse, de tal forma que, cuando está activado, el uso de marcadores realiza únicamente con el rayo.</p> <p>De la misma forma, cuando el rayo está desactivado, es el retículo el que determina la posición en el uso de marcadores.</p>
<b>Componentes</b>	CO-01, CO-02, CO-03, CO-06, CO-07
<b>Requisitos</b>	RS-F17

TABLA 8.26: CASO DE PRUEBA CP-25

<b>CP-25</b>	
<b>Objetivo</b>	Verificar la eliminación de todos los marcadores propios
<b>Entorno</b>	Realidad virtual, un usuario
<b>Acciones</b>	<p>Iniciar la aplicación eligiendo entorno con realidad virtual.</p> <p>Añadir varios marcadores.</p> <p>Mantener pulsado el gatillo de índice derecho de Oculus Touch durante un segundo para eliminar todos los marcadores.</p>
<b>Resultado esperado</b>	Se eliminan todos los marcadores del mapa.
<b>Componentes</b>	CO-01, CO-03, CO-06, CO-07, CO-11, CO-12
<b>Requisitos</b>	RS-F27



TABLA 8.27: CASO DE PRUEBA CP-26

<b>CP-26</b>	
<b>Objetivo</b>	Verificar la interacción individual con el mapa
<b>Entorno</b>	Ambos entornos, dos usuarios
<b>Acciones</b>	Los usuarios inician la aplicación, cada uno elige un entorno diferente. Cada usuario interacciona con su mapa (desplazamiento y <i>zoom</i> ).
<b>Resultado esperado</b>	Los dos usuarios pueden observar distintos puntos del mapa y con distinto nivel de <i>zoom</i> . Además, la interacción de un usuario con su mapa no afecta al mapa del otro usuario.
<b>Componentes</b>	CO-01, CO-03, CO-06, CO-07
<b>Requisitos</b>	RS-F09

TABLA 8.28: CASO DE PRUEBA CP-27

<b>CP-27</b>	
<b>Objetivo</b>	Verificar la colaboración en adición de marcadores
<b>Entorno</b>	Ambos entornos, dos usuarios
<b>Acciones</b>	Los usuarios inician la aplicación, cada uno elige un entorno diferente. Los usuarios añaden marcadores al mapa.
<b>Resultado esperado</b>	Cuando un usuario añade un marcador a su mapa, también se añade al mapa que observa el otro usuario. Cada usuario crea los marcadores de un color distinto al que poseen los del otro usuario. Los mapas que observan los dos usuarios son equivalentes en cuanto a número, posición y color de marcadores.
<b>Componentes</b>	CO-01, CO-03, CO-06, CO-07, CO-11, CO-12
<b>Requisitos</b>	RS-F13, RS-F25, RS-F28, RS-F39

TABLA 8.29: CASO DE PRUEBA CP-28

<b>CP-28</b>	
<b>Objetivo</b>	Verificar la colaboración en eliminación de marcadores
<b>Entorno</b>	Ambos entornos, dos usuarios
<b>Acciones</b>	Los usuarios inician la aplicación, cada uno elige un entorno diferente. Los usuarios añaden marcadores al mapa. Cada usuario intenta eliminar algunos marcadores situados en el mapa.
<b>Resultado esperado</b>	Si el usuario no ha creado el marcador, este no se elimina, e indica el error con un sonido. Si el usuario ha creado el marcador, este se elimina en su mapa. Además, también se elimina en el mapa que observa el otro usuario.
<b>Componentes</b>	CO-01, CO-03, CO-06, CO-07, CO-11, CO-12
<b>Requisitos</b>	RS-F11, RS-F12, RS-F15, RS-F33

TABLA 8.30: CASO DE PRUEBA CP-29

<b>CP-29</b>	
<b>Objetivo</b>	Verificar el límite de usuarios
<b>Entorno</b>	Ambos entornos, nueve usuarios
<b>Acciones</b>	Los nueve usuarios inician la aplicación (incluye introducir alias y pulsar el botón <i>Start</i> ) de forma ordenada.
<b>Resultado esperado</b>	El último usuario que inicie la aplicación no observará el mapa, ya que no pasará de la pantalla de conexión.
<b>Componentes</b>	CO-10, CO-12
<b>Requisitos</b>	RS-F29, RS-F30, RS-F31

TABLA 8.31: CASO DE PRUEBA CP-30

<b>CP-30</b>	
<b>Objetivo</b>	Verificar la colaboración en eliminación de todos los marcadores propios
<b>Entorno</b>	Ambos entornos, dos usuarios
<b>Acciones</b>	<p>Los usuarios inician la aplicación, cada uno elige un entorno diferente.</p> <p>Los usuarios añaden marcadores al mapa.</p> <p>El usuario 1 elimina todos los marcadores que ha creado de forma simultánea.</p> <p>El usuario 2 elimina todos los marcadores que ha creado de forma simultánea.</p>
<b>Resultado esperado</b>	Se respeta el orden de eliminación de los marcadores, desapareciendo primero aquellos pertenecientes al primer usuario que realizó la acción (usuario 1).
<b>Componentes</b>	CO-01, CO-03, CO-06, CO-07, CO-11, CO-12
<b>Requisitos</b>	RS-F11, RSF12, RS-F27

TABLA 8.32: CASO DE PRUEBA CP-31

<b>CP-31</b>	
<b>Objetivo</b>	Verificar la adición de marcadores al entrar en la sala
<b>Entorno</b>	Ambos entornos, dos usuarios
<b>Acciones</b>	<p>El usuario 1 inicia la aplicación, eligiendo un entorno.</p> <p>El usuario 1 añade marcadores.</p> <p>El usuario 2 inicia la aplicación, eligiendo el otro entorno.</p>
<b>Resultado esperado</b>	El usuario 2 observa en el mapa todos los marcadores que ha añadido el usuario 1.
<b>Componentes</b>	CO-03, CO-06, CO-07, CO-11, CO-12
<b>Requisitos</b>	RS-F32

TABLA 8.33: CASO DE PRUEBA CP-32

<b>CP-32</b>	
<b>Objetivo</b>	Verificar la eliminación de marcadores al salir de la sala
<b>Entorno</b>	Ambos entornos, dos usuarios
<b>Acciones</b>	Los usuarios inician la aplicación, cada uno elige un entorno diferente. El usuario 1 añade marcadores. El usuario 2 inicia la aplicación, eligiendo el otro entorno. El usuario 1 cierra la aplicación.
<b>Resultado esperado</b>	Todos los marcadores que había añadido el usuario 1 desaparecen del mapa del usuario 2.
<b>Componentes</b>	CO-03, CO-06, CO-07, CO-11, CO-12
<b>Requisitos</b>	RS-F34

TABLA 8.34: CASO DE PRUEBA CP-33

<b>CP-33</b>	
<b>Objetivo</b>	Verificar el destaque de marcadores al entrar la sala
<b>Entorno</b>	Ambos entornos, dos usuarios
<b>Acciones</b>	El usuario 1 inicia la aplicación, eligiendo un entorno. El usuario 1 añade marcadores. El usuario 1 destaca un marcador. El usuario 2 inicia la aplicación, eligiendo el otro entorno.
<b>Resultado esperado</b>	El usuario 2 observa en el mapa todos los marcadores que ha añadido el usuario 1. El marcador está destacado.
<b>Componentes</b>	CO-03, CO-06, CO-07, CO-11, CO-12
<b>Requisitos</b>	RS-F36

TABLA 8.35: CASO DE PRUEBA CP-34

<b>CP-34</b>	
<b>Objetivo</b>	Verificar la colaboración en destaque de marcadores
<b>Entorno</b>	Ambos entornos, dos usuarios
<b>Acciones</b>	<p>Los usuarios inician la aplicación, cada uno elige un entorno diferente.</p> <p>Los usuarios añaden marcadores.</p> <p>El usuario 1 destaca un marcador creado por el usuario 2.</p> <p>El usuario 2 destaca el mismo marcador que acaba de destacar el usuario 1.</p> <p>El usuario 1 desplaza el mapa de tal forma que no sea visible el marcador destacado.</p>
<b>Resultado esperado</b>	<p>Cuando un usuario destaca un marcador en su mapa, se destaca el mismo marcador en el mapa que observa el otro usuario.</p> <p>Cuando el usuario 2 destaca el marcador, el color de destaque de este es el asociado al usuario 2.</p> <p>Tras realizar todas las acciones, el color del marcador y su color de destaque no coinciden, y se distingue a través de estos colores quién es el usuario creador y el usuario que lo ha destacado.</p> <p>Al finalizar, el usuario 1 observa una flecha que señala el lugar donde se encuentra el marcador destacado. El color de esta flecha es el color asociado al usuario 2.</p>
<b>Componentes</b>	CO-01, CO-03, CO-06, CO-07, CO-11, CO-12
<b>Requisitos</b>	RS-F20, RS-F22, RS-F23, RS-F26, RS-F35, RS-F38

TABLA 8.36: CASO DE PRUEBA CP-35

<b>CP-35</b>	
<b>Objetivo</b>	Verificar el chat de voz
<b>Entorno</b>	Ambos entornos, dos usuarios
<b>Acciones</b>	<p>Los usuarios inician la aplicación, el usuario 1 elige entorno sin realidad virtual, y el usuario 2 el entorno con realidad virtual.</p> <p>Los usuarios hablan.</p> <p>El usuario 1 desactiva y vuelve a activar el micrófono pulsando la tecla 'M' del teclado.</p> <p>El usuario 2 desactiva y vuelve a activar el micrófono pulsando el botón 'Y' del Oculus Touch izquierdo.</p>
<b>Resultado esperado</b>	<p>El micrófono está activado por defecto.</p> <p>Cuando el micrófono de un usuario está activado, el otro usuario escucha lo que este dice.</p> <p>Cuando el micrófono de un usuario está desactivado, el usuario no escucha lo que este dice.</p>
<b>Componentes</b>	CO-13, CO-14, CO-12
<b>Requisitos</b>	RS-F40, RS-F41, RS-F42

TABLA 8.37: CASO DE PRUEBA CP-36

<b>CP-36</b>	
<b>Objetivo</b>	Verificar la lista de usuarios en sala
<b>Entorno</b>	Ambos entornos, dos usuarios
<b>Acciones</b>	<p>Los usuarios inician la aplicación, el usuario 1 elige entorno sin realidad virtual, y el usuario 2 el entorno con realidad virtual.</p> <p>Los usuarios añaden marcadores.</p> <p>Los usuarios se comunican por voz.</p> <p>El usuario 1 oculta y vuelve a mostrar la lista de usuarios pulsando la tecla 'TAB' del teclado.</p> <p>El usuario 2 oculta y vuelve a mostrar la lista de usuarios pulsando el botón 'X' del Oculus Touch izquierdo.</p>
<b>Resultado esperado</b>	<p>Cada usuario puede observar una lista de usuarios, en la que se muestra su alias y su color asociado. El color corresponde con el color de los marcadores que crean, y el color de destaque de los marcadores que destacan.</p> <p>Cada vez que un usuario transmite audio, se muestra el icono que lo indica en la lista de jugadores.</p> <p>Cuando un usuario silencia el micrófono, se muestra el icono que lo indica en la lista de jugadores.</p> <p>La lista de jugadores se puede ocultar y volver a mostrar.</p>
<b>Componentes</b>	CO-01, CO-05, CO-09, CO-12
<b>Requisitos</b>	RS-F45, RS-F46, RS-F47, RS-F48

TABLA 8.38: CASO DE PRUEBA CP-37

<b>CP-37</b>	
<b>Objetivo</b>	Verificar las flechas que señalan marcadores destacados
<b>Entorno</b>	Realidad virtual, un usuario
<b>Acciones</b>	<p>Iniciar la aplicación eligiendo entorno sin realidad virtual.</p> <p>Añadir un marcador.</p> <p>Destacar el marcador añadido.</p> <p>Desplazar el mapa, de tal forma que el marcador destacado desaparezca de la pantalla.</p>
<b>Resultado esperado</b>	Mientras el marcador destacado es visible, no se muestran las flechas. Cuando el marcador deja de ser visible, las flechas se muestran señalando la posición de este. El color de la flecha coincide con el del círculo que rodea el marcador señalado.
<b>Componentes</b>	CO-01, CO-03, CO-04, CO-08
<b>Requisitos</b>	RS-F37

TABLA 8.39: CASO DE PRUEBA CP-38

<b>CP-38</b>	
<b>Objetivo</b>	Prueba final
<b>Entorno</b>	Ambos entornos, ocho usuarios
<b>Acciones</b>	<p>Los usuarios inician la aplicación, un usuario elige entorno con realidad virtual, y el resto de los usuarios el entorno sin realidad virtual.</p> <p>Los usuarios utilizan la aplicación, aprovechando todas las funcionalidades disponibles, durante al menos 10 minutos.</p>
<b>Resultado esperado</b>	No se detecta ningún error de funcionalidad.
<b>Componentes</b>	Todos los componentes
<b>Requisitos</b>	Todos los requisitos



## 8.2. Resultados de las pruebas

Los resultados de las pruebas se muestran a través de una tabla, en la cual se indica el identificador del caso de prueba, si esta fue superada o no, y observaciones relativas a la prueba realizada. En el caso del apartado de observaciones, se describen los errores encontrados si la prueba no fue superada, además de posibles propuestas de mejora.

TABLA 8.40: RESULTADOS DE LAS PRUEBAS (I)

Caso de prueba	Prueba superada	Observaciones
CP-01	Sí	
CP-02	Sí	El mapa requiere cierto tiempo para su carga completa cuando que se cambia el nivel de zoom (puede frustrar al usuario si la interacción es demasiado rápida).  Con nivel de <i>zoom</i> mínimo no aparecen los marcadores en el mapa (parece problema de <i>OnlineMaps</i> ).
CP-03	Sí	
CP-04	Sí	
CP-05	Sí	
CP-06	Sí	
CP-07	Sí	
CP-08	Sí	
CP-09	Sí	
CP-10	Sí	
CP-11	Sí	La aplicación debe iniciarse con un micrófono conectado. En caso contrario, se da un error de ejecución.
CP-12	Sí	Ver CP-02.
CP-13	Sí	El mapa vuelve a la posición inicial durante unos milisegundos antes de completar el desplazamiento.

TABLA 8.41: RESULTADOS DE LAS PRUEBAS (II)

<b>Caso de prueba</b>	<b>Prueba superada</b>	<b>Observaciones</b>
CP-14	Sí	
CP-15	Sí	
CP-16	Sí	
CP-17	Sí	
CP-18	Sí	
CP-19	Sí	
CP-20	Sí	
CP-21	Sí	
CP-22	Sí	
CP-23	Sí	
CP-24	Sí	
CP-25	Sí	
CP-26	Sí	
CP-27	Sí	
CP-28	Sí	
CP-29	No	La aplicación deja que el usuario se conecte. Provoca que el usuario no tenga color asociado y la aplicación sufra un error de ejecución.
CP-30	Sí	
CP-31	Sí	
CP-32	Sí	
CP-33	Sí	
CP-34	Sí	

TABLA 8.42: RESULTADOS DE LAS PRUEBAS (III)

Caso de prueba	Prueba superada	Observaciones
CP-35	Sí	
CP-36	Sí	La lista no se observa en el entorno de realidad virtual, pero sí en el monitor del ordenador. Los elementos gráficos utilizados no son compatibles con entornos de realidad virtual.
CP-37	Sí	Las flechas pueden obstaculizar la observación de otros marcadores, aunque estas parpadeen.
CP-38	No	Pérdida de rendimiento en el HMD cuando existen 8 usuarios simultáneos. Al girar el cuello, existe un retardo a la hora de cargar la imagen.

La mayoría de las pruebas han sido superadas. Se han encontrado errores leves en casos de prueba que se han superado, los cuales tendrán en cuenta en versiones posteriores de la aplicación. Los casos de prueba no superados fueron dos:

- **CP-29.** Se comprobaba que se limitaba el número de usuarios en la sala a 8. Se desconocen las razones por las que el noveno usuario pudo conectarse, ya que se definió en el código el máximo número de jugadores a 8. Es posible que sea un dato que deba tener en cuenta el desarrollador en la implementación, y no sea *Photon* el que se encargue de controlar el número de conexiones. Asimismo, se trata de un error menor, que no influye en los objetivos del proyecto.
- **CP-38.** Se trataba de la prueba final, en la que se probaba la aplicación con 8 usuarios simultáneos durante al menos 10 minutos. Se observó una gran pérdida de rendimiento cuando se utilizaba el HMD Oculus Rift. En este caso, sí se trata de un aspecto que puede afectar a los objetivos del proyecto.

La caída de rendimiento puede deberse a varios factores:

- **Limitaciones a nivel hardware.** Es posible que el procesador o la tarjeta gráfica no soportaran el tratamiento de información de los 8 usuarios de la sala, y afectara al rendimiento del HMD. Se recomienda probar la aplicación en un ordenador con mayor potencia de procesamiento.
- **Código poco optimizado.** Debido a que en Unity se da un gran número de funciones *Update*, en las que se realiza una tarea en cada fotograma, cabe la posibilidad de que el uso frecuente de este tipo de funciones provocara la caída de rendimiento cuando el número de usuarios era

elevado. En versiones posteriores, se analizará con detalle el código para encontrar las soluciones óptimas.

- **HMD sobrecargado.** Las pruebas se realizaron una tras otra, requiriendo el uso continuo del dispositivo. Al tratarse de la última prueba realizada, es posible que se tratara de una de las causas. Asimismo, al terminar todas las pruebas, el HMD presentaba una temperatura ligeramente superior a la habitual. En este caso, se trataría de un error temporal que podría ser asumible.

## 9. CONCLUSIONES

### 9.1. Objetivos cumplidos y conclusiones generales

Tras la finalización del proyecto, se analizan los objetivos planteados al inicio del mismo, con el fin de comprobar si estos se han cumplido o no.

El primer objetivo del proyecto consistía en desarrollar un entorno colaborativo de visualización de mapas en realidad virtual. Se trata del objetivo principal del proyecto, por lo que era el primero que era necesario cumplir. Es cierto que se detectaron problemas de rendimiento al realizar pruebas de ejecución de la aplicación con ocho usuarios simultáneos, pero este aspecto no afecta a la consecución del objetivo planteado, ya que se ha asegurado que el sistema funciona correctamente en escenarios con un menor número de usuarios. Además, no se puede afirmar con certeza que esta pérdida de rendimiento se deba a la eficiencia del código. Por tanto, este objetivo se ha cumplido, dado que se ha conseguido desarrollar una aplicación que cumple las características descritas.

En cuanto al segundo objetivo, era la extensión de la colaboración a entornos con ausencia de inmersión en un entorno de realidad virtual. Para conseguir esto, la aplicación ofrece dos tipos de entornos que pueden colaborar entre sí, y uno de ellos no precisa el uso de un HMD de realidad virtual. En ese caso, la imagen se muestra a través de un monitor convencional. Por tanto, este objetivo también se ha cumplido.

En cuanto al tercer objetivo, consistía en aprovechar la implementación en una nueva tecnología para analizar nuevas formas de interacción aplicables a los sistemas de visualización e interacción con mapas, el cual también se ha cumplido. Para ello, el entorno de realidad virtual permite dos formas de señalar una posición en el mapa. Una de ellas se basa en la utilización del centro de la mirada, mientras que la otra crea un rayo procedente del Oculus Touch Controller. La existencia simultánea de dos formas de interacción permite compararlas entre sí, de tal forma que se podrá conocer cuál es la alternativa más eficiente para este tipo de sistemas.

Tras analizar los objetivos descritos al inicio del proyecto, se puede concluir en que estos se han alcanzado con éxito.

Por otro lado, es importante destacar las conclusiones extraídas tras la finalización del proyecto, basándose en la experiencia de los últimos siete meses. Se ha tratado de un trabajo que ha requerido un esfuerzo considerable, sobre todo al comienzo del mismo, ya que ha sido necesario un tiempo adicional en aprender a utilizar la plataforma de desarrollo de Unity, un nuevo lenguaje de programación (C#), y comenzar a trabajar con nuevas tecnologías como es la realidad virtual, aspectos que no se encontraban entre las diferentes materias cursadas a lo largo del Grado. A pesar del esfuerzo realizado, no ha sido en vano, ya que se han adquirido nuevos conocimientos y experiencias que contribuyen a la formación de un ingeniero versátil, capaz de adaptarse a nuevas circunstancias, a pesar de salir de su zona de confort.

Además, tras la experiencia en desarrollo y uso de realidad virtual, se puede afirmar que no es tan complejo como parece en un inicio, y es evidente que esta tecnología posee una gran proyección de futuro. Ofrece experiencias únicas al usuario, lo cual puede ser

aprovechado para dar soporte a todo tipo de actividades en diferentes áreas, incluso mejorando las mismas.

Del mismo modo, al poseer una gran proyección de futuro, también existen aspectos que se pueden mejorar. En primer lugar, es posible distinguir los píxeles en las pantallas de los HMD, lo cual disminuye la inmersión y, por tanto, la presencia. Para evitar esto, debería proporcionar una mayor resolución en sus pantallas.

También, es necesario considerar la incomodidad que puede producir el uso de cables, así como el tamaño del visor, que engloba la cabeza al completo, lo cual puede no ser soportable en épocas de calor extremo. En este caso, los HMDs deberían reducir su tamaño o, al menos, mejorar su ventilación. En cuanto al uso de cables, ya están comenzando a comercializarse HMDs inalámbricos, que dotan de mayor libertad de movimiento al sistema.

## **9.2. Futuras líneas de trabajo**

El presente trabajo sirve como punto de partida para aplicar la tecnología de realidad virtual a entornos de visualización e interacción con mapas. Por ello, se consideran futuras líneas de trabajo, ya sea para ofrecer otras funcionalidades o mejorar las existentes.

En primer lugar, cabe considerar los resultados obtenidos tras la realización de las pruebas sobre la aplicación desarrollada, de cara a su perfeccionamiento. En este aspecto, la prioridad es detectar la causa de la pérdida de rendimiento al alcanzar el máximo de ocho usuarios simultáneos. En caso de que el problema residiera en la implementación, se adoptarían alternativas más eficientes a nivel de procesamiento.

Después de tratar esta cuestión, se procedería a acabar con los errores menores, tales como la ocultación de los marcadores cuando el mapa ofrece su nivel de *zoom* mínimo, o los problemas de fluidez al desplazar el mapa en el entorno de realidad virtual.

En segundo lugar, para comprobar la efectividad de las formas de interacción implementadas, se podrían realizar experimentos en los que varios usuarios con experiencia en entornos de visualización e interacción con mapas utilicen la aplicación desarrollada. El objetivo de estos experimentos es conocer cuál es la alternativa más eficiente para este tipo de sistemas.

Debido al tiempo limitado del que se disponía para el desarrollo del proyecto, no se contemplaron otras formas de interacción que podían resultar útiles para el ámbito de la visualización e interacción con mapas. Por ello, una posible línea de trabajo sería la adición de nuevas formas de interacción basadas en tecnología *leap motion*, la cual permite la interacción a través de movimientos realizados con las manos. Se trata de una alternativa que favorece la inmersión y la idea de presencia.

Otra línea de trabajo consistiría en utilizar un plano curvado para mostrar el mapa, de tal forma que la distancia entre el usuario y todos los puntos del mapa sea constante. Con esto, se consigue que el usuario observe todos los elementos del mapa con el mismo tamaño, lo cual permitirá la utilización de mapas de mayor tamaño, así como una mayor inmersión del usuario.

Por último, se propone la creación de un menú de configuración en el que el usuario pueda personalizar los controles de juego, eligiendo el botón asociado a cada una de las acciones disponibles, así como la posibilidad de crear varias salas para admitir varios equipos de trabajo.

## 10.GESTION DEL PROYECTO

### 10.1. Planificación

En la figura 10.1 se muestra un diagrama de Gantt que representa la planificación inicial del proyecto. Se trata de una planificación semanal, con el objetivo de finalizar el proyecto en 25 semanas. Aunque el límite de entrega se encuentre en la segunda semana de junio (semana 24 del diagrama), se planea la finalización una semana antes. En caso de producirse algún retraso en la realización del proyecto, ya sea por el suceso de un imprevisto o un error en la planificación, se dispondrá de una semana adicional para poder cumplir el límite de entrega.

Se estimó una media de 10 horas semanales de trabajo, por lo que el proyecto requerirá un trabajo total de 250 horas.

Cabe destacar que, debido a que era la primera vez que se desarrollaba en la plataforma de desarrollo de Unity, se decidió dedicar un mes al aprendizaje, de tal forma que se tuviera el conocimiento suficiente sobre la plataforma para una mejor toma de decisiones en las fases de análisis y diseño. Asimismo, se evitarán modificaciones innecesarias.

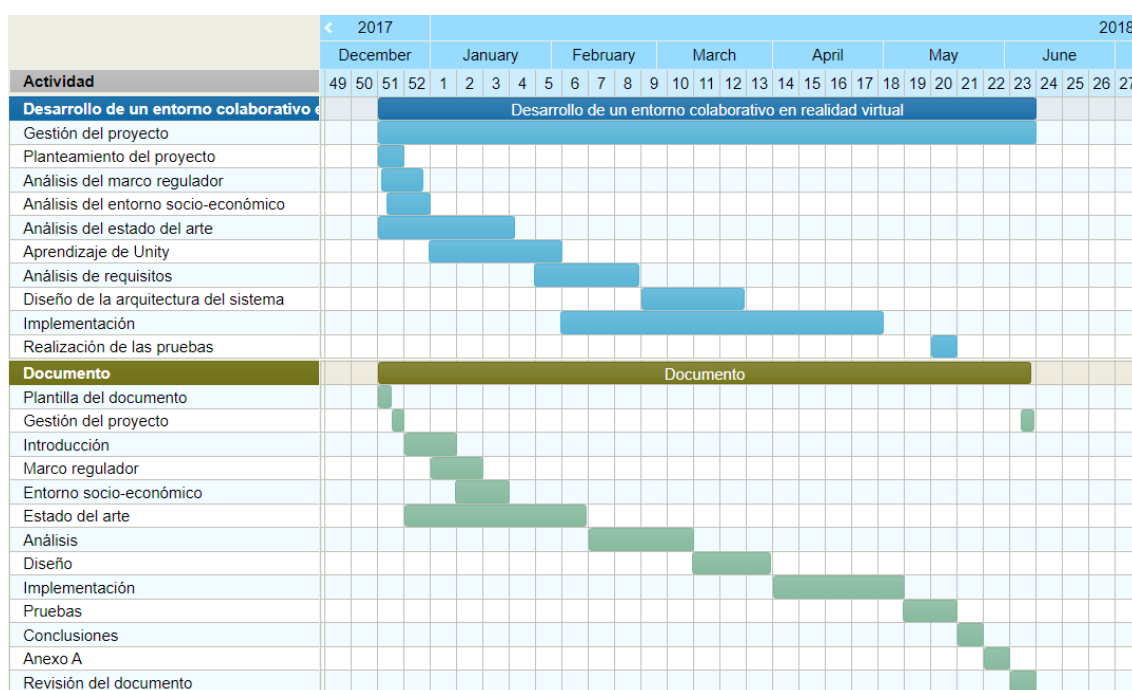


Fig. 10.1. Diagrama de Gantt inicial del proyecto

En cuanto a la ejecución de la planificación anterior, se representa en el diagrama de Gantt de la figura 10.2. Cabe destacar que no se esperaba tanta carga de trabajo en el resto de las asignaturas del curso. Esto produjo un retraso en las primeras tareas, entre las que se incluye el aprendizaje de Unity. Para solventar este problema, fue necesario comenzar la implementación mientras se finalizaba la fase de aprendizaje, lo cual implicaba una mayor dedicación semanal en el proyecto.



Además, a la hora de realizar la planificación, no se tuvieron en cuenta los exámenes finales, para los que fue necesaria una dedicación exclusiva durante las dos primeras semanas de mayo, y el correspondiente retraso en la realización del proyecto. No obstante, fue posible la finalización antes del límite de entrega, gracias a la disponibilidad de una semana adicional.

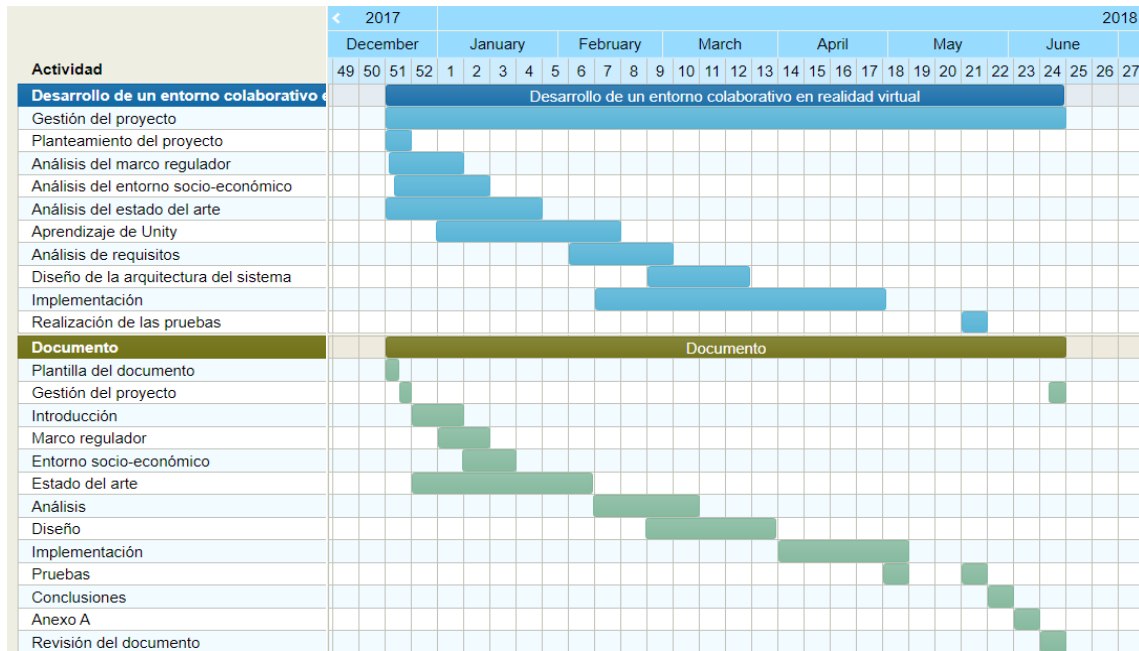


Fig. 10.2. Diagrama de Gantt final del proyecto

Por tanto, el proyecto duró una semana más de lo estimado. Asimismo, debido a los retrasos mencionados, fue necesario dedicar un mayor número de horas al proyecto, ascendiendo a un total de 275 horas.

## 10.2. Método de desarrollo

El método de desarrollo utilizado para la realización del proyecto de desarrollo de software es el modelo en cascada con retroalimentación. Tal y como se observa en la figura 10.3, se divide en cinco fases:

- **Análisis de requisitos.** Se realiza la especificación de requisitos del sistema, en la que se definen de forma exhaustiva sus funcionalidades y restricciones.
- **Diseño del sistema.** Se detalla la arquitectura del sistema, realizando una división en diferentes módulos o componentes. Se establece la función de cada componente y las relaciones existentes entre estos.
- **Implementación.** Codificación y construcción de cada uno de los módulos
- **Pruebas.** Se verifica el correcto funcionamiento de la aplicación.
- **Mantenimiento.** Corrección de errores o mejora del rendimiento del software tras su entrega. Esta fase no se recoge en el documento.

Existe retroalimentación entre las fases, de tal forma que, si se detecta un error en cualquiera de las ellas, es posible volver a una etapa anterior para solventar el problema.

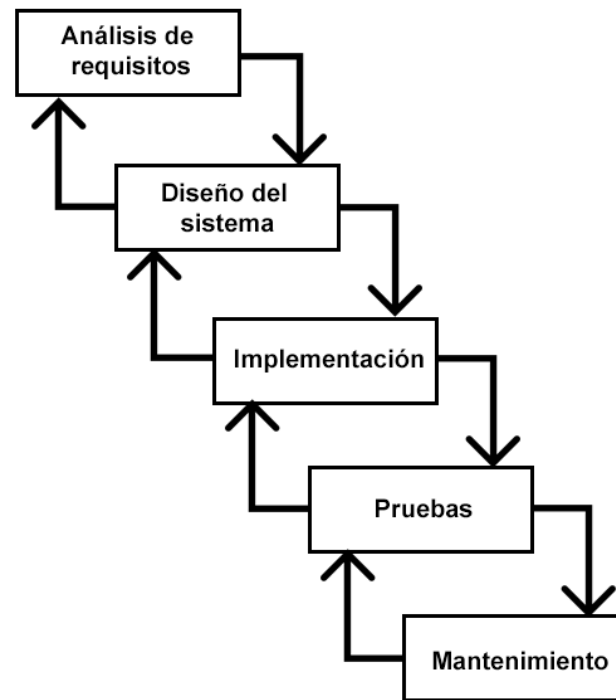


Fig. 10.3. Modelo en cascada con retroalimentación

## 11.BIBLIOGRAFÍA

- [1] “Usuarios de realidad virtual pagada 2015-2018 | Estadística”, *Statista*, jun. 2014. [En línea]. Disponible en: <https://es.statista.com/estadisticas/599952/usuarios-de-realidad-virtual-pagada-en-el-mundo--2018/>
- [2] L. Ying, Z. Jiong, S. Wei, W. Jingchun, y G. Xiaopeng, “VREX: Virtual reality education expansion could help to improve the class experience (VREX platform and community for VR based education)”, presentada en 2017 IEEE Frontiers in Education Conference (FIE), Indianápolis, 18-21 oct., 2017, pp. 1-5. [En línea]. Disponible en: <http://doi.org/10.1109/FIE.2017.8190660>
- [3] R. Mourning, y T. Ying, “Virtual reality social training for adolescents with high-functioning autism”, presentada en 2016 IEEE International Conference on Systems, Man, and Cybernetics, SMC 2016 - Conference Proceedings, Budapest, 9-12 oct., 2016, pp. 4848-4853. [En línea]. Disponible en: <http://doi.org/10.1109/SMC.2016.7844996>
- [4] S. Wijewickrema et al., “Design and Evaluation of a Virtual Reality Simulation Module for Training Advanced Temporal Bone Surgery”, presentada en 2017 IEEE 30th International Symposium on Computer-Based Medical Systems (CBMS), Tesalónica, 22-24 jun., 2017, pp. 7-12. [En línea]. Disponible en: <http://doi.org/10.1109/CBMS.2017.10>
- [5] “Oculus Rift | Oculus”, *Oculus*. [En línea]. Disponible en: <https://www.oculus.com/legal/health-and-safety-warnings/>
- [6] “Khronos Announces VR Standards Initiative - The Khronos Group Inc”, *Khronos*, 06-12-2016. [En línea]. Disponible en: <https://www.khronos.org/news/press/khronos-announces-vr-standards-initiative>
- [7] “OpenXR Overview - The Khronos Group Inc”, *Khronos*, 06-12-2016. [En línea]. Disponible en: <https://www.khronos.org/openxr>
- [8] D. A. Bowman, y R. P. McMahan, “Virtual Reality: How Much Immersion Is Enough?”, *Computer*, vol. 40, n.º 7, pp. 36-43, 2007. [En línea]. Disponible en: <http://doi.org/10.1109/MC.2007.257>
- [9] M. Slater, “A Note on Presence Terminology”, *Presence Connect*, vol. 3, 2003. [En línea]. Disponible en: [https://www.researchgate.net/publication/242608507\\_A\\_Note\\_on\\_Presence\\_Terminology](https://www.researchgate.net/publication/242608507_A_Note_on_Presence_Terminology)
- [10] D. Horváthová, D., V. Siládi, V., y E. Lacková, “Phobia treatment with the help of virtual reality”, presentada en 2015 IEEE 13th International Scientific Conference on Informatics, Poprad, 18-20 nov., 2015. [En línea]. Disponible en: <http://doi.org/10.1109/Informatics.2015.7377818>
- [11] I. Maach, A. Azough, y M. Meknassi, “Development of a use case for virtual reality to visit a historical monument”, presentada en 2018 International Conference on Intelligent Systems and Computer Vision (ISCV), Fez-Marruecos, 2-4 abr., 2018, pp. 1–4. [En línea]. Disponible en: <https://doi.org/10.1109/ISACV.2018.8354052>

- [12] R. Arnaldo, L. Perez, J. Crespo, y F. Alonso, "The use of virtual flight simulation for airspace design in aeronautical engineering education", presentada en 2011 IEEE Global Engineering Education Conference (EDUCON), Amán, 4-6 abr., 2011, pp.543-551. [En línea]. Disponible en: <http://doi.org/10.1109/EDUCON.2011.5773191>
- [13] A. Mühlberger, M. J. Herrmann, G. Wiedemann, H. Ellgring, y P. Pauli, "Repeated exposure of flight phobics to flights in virtual reality", *Behav. Res. Ther.*, vol. 39, n.º 9, pp. 1033–1050, 2001. [En línea]. Disponible en: [https://doi.org/10.1016/S0005-7967\(00\)00076-0](https://doi.org/10.1016/S0005-7967(00)00076-0)
- [14] M. Krijn, P. M. G. Emmelkamp, R. Biemond, C. deWilde de Ligny, M. J. Schuemie, and C. A. P. G. van der Mast, "Treatment of acrophobia in virtual reality: The role of immersion and presence", *Behav. Res. Ther.*, vol. 42, n.º 2, pp. 229–239, 2004. [En línea]. Disponible en: <https://www.sciencedirect.com/science/article/pii/S0005796703001396?via%3Dihub>
- [15] S. Gupta y B. Chadha, "Virtual reality against addiction", presentada en International Conference on Computing, Communication & Automation, Greater Noida, 15-16 may., 2015, pp. 273–278. [En línea]. Disponible en: <https://doi.org/10.1109/CCAA.2015.7148387>
- [16] J. Elliman, M. Loizou, y F. Loizides, "Virtual Reality Simulation Training for Student Nurse Education", presentada en 2016 8th International Conference on Games and Virtual Worlds for Serious Applications (VS-GAMES), 2016, pp. 1–2. [En línea]. Disponible en: <https://doi.org/10.1109/VS-GAMES.2016.7590377>
- [17] C. Suárez-Mejías, G. G. Ciriza, C. P. Calderón, T. G. Cía, y P. Gacto-Sánchez, "Virtual reality simulation training and assisted surgery: AYRA: Virtual and physical biomodels in surgery", presentada en 2012 18th International Conference on Virtual Systems and Multimedia, Milán, 2-5 sep., 2012, pp. 437–444. [En línea]. Disponible en: <https://doi.org/10.1109/VSMM.2012.6365956>
- [18] D. E. Holmes, D. K. Charles, P. J. Morrow, S. McClean, y S. M. McDonough, "Using Fitt's Law to Model Arm Motion Tracked in 3D by a Leap Motion Controller for Virtual Reality Upper Arm Stroke Rehabilitation", presentada en 2016 IEEE 29th International Symposium on Computer-Based Medical Systems (CBMS), Dublín y Belfast, 20-23 jun., 2016, pp. 335–336. [En línea]. Disponible en: <https://doi.org/10.1109/CBMS.2016.41>
- [19] W. Hürst, W., B. Boer, B. de, W. Florijn, W., y X. J. Tan, "Creating new museum experiences for virtual reality", presentada en 2016 IEEE International Conference on Multimedia & Expo Workshops (ICMEW), Seattle, 11-15 jun., 2016. [En línea]. Disponible en: <http://doi.org/10.1109/ICMEW.2016.7574692>
- [20] F. Bruno et al., "From 3d reconstruction to virtual reality: A complete methodology for digital archaeological exhibition", *Journal of Cultural Heritage*, vol. 11, n.º. 1, pp. 42–49, 2010. [En línea]. Disponible en: [https://www.researchgate.net/publication/248546261\\_From\\_3D\\_reconstruction\\_to\\_virtual\\_reality\\_A\\_complete\\_methodology\\_for\\_digital\\_archaeological\\_exhibition](https://www.researchgate.net/publication/248546261_From_3D_reconstruction_to_virtual_reality_A_complete_methodology_for_digital_archaeological_exhibition)

[21] S. Pick, S. Gebhardt, B. Weyers, B. Hentschel, y T. Kuhlen, "A 3D collaborative virtual environment to integrate immersive virtual reality into factory planning processes", presentada en 2014 International Workshop on Collaborative Virtual Environments (3DCVE), Minneapolis, 30 mar., 2014, pp. 1–6. [En línea]. Disponible en: <https://doi.org/10.1109/3DCVE.2014.7160934>

[22] S. Cailhol, P. Fillatreau, J. Y. Fourquet, y Y. Zhao, "A Multi-layer Approach for Interactive Path Planning Control in Virtual Reality Simulation", presentada en 2014 International Conference on Virtual Reality and Visualization, Shenyang Aerospace University, 30-31 ago., 2014, pp. 296–301. [En línea]. Disponible en: <https://doi.org/10.1109/ICVRV.2014.76>

[23] "Minimum Specs – Oculus Support", *Oculus*. [En línea]. Disponible en: <https://support.oculus.com/170128916778795/>

## **12. DEVELOPMENT OF A COLLABORATIVE VIRTUAL REALITY ENVIRONMENT**

### **12.1. Introduction**

#### **12.1.1. Context**

Over the last few years, there has been an important technological development, which has caused the appearance of many devices where virtual environments are generated. In case of a completely virtual environment, it is known as Virtual Reality (VR).

This project is focused in development using virtual reality technology. It offers an experience in which users are immersed in a computed-generated environment. This is the main difference with augmented reality, which adds virtual elements to a real environment.

The main component for the generation of these virtual environments are helmets or glasses, also known as Head-Mounted Displays (HMD). These glasses could be a support to introduce a smartphone or, in the case of the most sophisticated ones, they have a built-in screen. In the case of augmented reality, due to the virtualization of only a fraction of the environment, it is common to use the smartphone without placing it on a support. However, there are models of augmented reality glasses with a built-in screen, for example, the Microsoft HoloLens.

Currently, the use of built-in screen glasses is greater in the case of virtual reality than in augmented reality. This is because there is a lower number of augmented reality HMDs, mostly versions for developers. Therefore, its price is considerably higher than virtual reality glasses, which offer more versions for consumers.

VR built-in screen glasses are beginning to be economically possible for average users. Virtual reality technology has been gaining greater importance in the market recently, as it is increasing the number of companies that offer their own device to enjoy a virtual reality experience. It has been estimated that in 2018 there will be at least 28 million virtual reality users [1].

It is a fact that the main use that is being given to this technology is associated with the video game industry. However, there are also applications in education, the therapeutic field or the professional environment.

It is important to take advantage of the possibilities that new technologies can offer, so they can be used when they are more affordable. Therefore, it is positive to experiment with new interaction paradigms, for facilitating the performance of tasks in all possible contexts. Virtual reality is a technology that still has room for improvement, so it can offer endless possibilities for the future.

### **12.1.2. Motivation of the project**

The main motivation of this work is to explore the possibilities that virtual reality technology has to offer in collaborative environments, in this case, it is applied to map interaction and visualization environments.

Virtual reality technology has been gaining greater importance in the market recently, as it is increasing the number of companies that offer their own device to enjoy a virtual reality experience. Virtual reality devices are reducing their prices, so they are becoming economically affordable for the average user. Although it is not a new technology, it is increasing its use, as it facilitates the acquisition of its devices for domestic usage or, at least, outside the scope of laboratories and large companies. Therefore, it is time to explore the opportunities offered by the application of this technology, as well as the potential improvement of tasks in different situations.

This project is focused on the usage of virtual reality technology to offer a lower-priced alternative for collaborative map visualization and interaction environments, such as emergency coordination centers. These centers frequently utilize large displays, which are usually overpriced, to afford the collaboration between team members. To solve this issue, it is proposed a virtual reality approach as an alternative of these large displays to reduce costs. In addition, it will allow collaboration between team members even though they are in different geographical locations.

Among the advantages of virtual reality for map visualization and interaction environments are the wide field of vision offered to the user, as well as the possibility of creating large virtual environments with an interaction made from a confined physical space. This allows to have in the virtual world a large canvas to dispose all the information, without spatial limitations. In addition, the user is immersed in the virtual environment without distractions, which facilitates and speeds up the completion of tasks. Therefore, it is important to highlight, in addition to the potential impact at the economic level, the usefulness that can be offered in the business field with the realization of this project.

On the other hand, this project offered the possibility of developing with a novel technology with a great projection of the future, such as virtual reality. It should be considered that this technology had not been worked on along the Degree. The same applies to using Unity as a development platform.

For this reason, the work required a considerable initial effort since it was necessary to learn to use a new tool, in addition to an adaptation period for this new technology. However, this project was presented as an opportunity to acquire new skills, as well as apply the concepts learned in the last few years. Also, it is an additional preparation that could be positive for the future.

### 12.1.3. Objectives

The main goals of this project are:

- **Development of a collaborative map visualization and interaction environment using virtual reality technology.** It is the main goal of the project. Several users are simultaneously immersed in a virtual environment in which a map is displayed. Through the interaction with that map, they will be able to communicate with each other. It is proposed an alternative that requires lower cost and physical space than the large screens currently used in emergency coordination centers.
- **Extension of collaboration to non-VR environments.** It implies that users can participate in the collaboration, even if they do not have virtual reality glasses. In this case, it is proposed the use of a current monitor and an interaction with the map through mouse and keyboard. Therefore, a user who uses virtual reality glasses and another who uses a monitor to view the map can communicate with each other. In fact, it is offered the opportunity to use a system based on virtual reality, not as a replacement, but as a compliment for the existing technology.
- **Analysis of effective forms of interaction.** Despite not being the main objective of the work, the development of this project will allow the study of new forms of interaction, applicable to map visualization and interaction systems. Virtual reality is a technology that is beginning in development, so there is hardly any information about it. The developed application will be useful to verify the effectiveness of the different interaction alternatives that virtual reality has to offer for this type of systems.

To achieve these objectives, it is necessary to start a software development project. In this project, it will be developed an application based on virtual reality, in which different users can collaborate through the interaction with a map. Each user can add, delete and highlight markers to a map, and other users will perceive each of these actions. In addition, they can communicate through a voice chat.



## 12.2. Methods

### 12.2.1. Virtual reality

Virtual reality is a type of computer-mediated reality which consists in the perception of a totally virtual environment as if it was the real world, that is, it is the user's immersion in a computer-generated reality.

The main property of this technology is known as immersion, which consists in provide, through a screen, an illusion of reality in their users. It can reach the point where there is a total immersion, in which user believes that is physically located in a virtual environment. The total immersion is called presence.

Nowadays, immersive environments are generated using HMDs. These devices are placed on the head, so the screen is placed in front of user's eyes and the virtual world is always observed. Likewise, it tracks head position and rotation thanks to the use of external or incorporated sensors.

The main advantages of virtual reality the following:

- **Immersion and presence.** Virtual reality is the only technology that can give the feeling of being in a different place and could reach the point where there is a total immersion, in which user believes that is physically located in a virtual environment.
- **Simulation.** It allows to practice in a totally controlled environment without taking any risks.
- **Cost saving.** It is possible to create any interactive space saving on costs of physical generation of the environment. This could reduce business mobility, which implies time and cost saving.
- **Entertainment.** The main use of virtual reality applications is related to leisure, due to its audiovisual approach. It brings new experiences to users that only this technology could offer.

There are other advantages that must be taken into consideration for this specific project, such as:

- **Task efficiency.** Virtual reality is based on immersive environments, without distractions, that can facilitate and speed up the completion of tasks. This fact may imply more efficiency in working environments. However, it requires an initial period in which employees will have to adapt to the use of these new devices. This period will not be necessary as soon as all people have a virtual reality kit at home, as expected, because of lower prices and the growth in the number of virtual reality users in the last few years.
- **Large virtual environments in confined physical spaces.** There are no spatial limitations in virtual worlds, so a large amount of information can be presented. Also, it requires a small physical space for executing applications where it is not necessary user's physical movement.

- **Portability of equipment.** Due to its size and modularity of the solution, equipment is easily carried and replaced in case of failure.
- **Different geographical locations.** This approach allows collaboration between team members regardless of their geographical locations.
- **Creation of different virtual environments.** Two different groups in a work team could collaborate in different virtual environments or the same one, based on the needs at any given time. Moreover, it will not affect in cost terms.

In the same way, there are some disadvantages of virtual reality technology:

- Its use can cause symptoms related to motion sickness: altered vision, disorientation, balance disturbances, dizziness, seizures, muscle or optic spasms, involuntary movements [5]. Fortunately, due to improvements in performance, these problems are decreasing.
- The weight of the HMD and the use of controllers can cause pain in muscles, articulations, neck, hands or skin [5].
- Possibility of transmitting contagious diseases because of the sharing of the equipment [5].
- User is immersed in a virtual environment, so its prolonged use can cause psychological problems from distortion of reality, to triggering or aggravating emotional disorders.
- Danger of hitting objects in the room and causing serious physical damage.
- User is not aware of what happens in the real world, due to the immersion. It could be dangerous in case of fires, landslides, burglaries...

Despite of these problems, if use recommendations are followed, it would not be an issue. In conclusion, there are many advantages to using virtual reality that could be positive in map interaction and visualization systems.

### **12.2.2. State of the art**

This section shows a summary of the alternatives chosen for the development of the project, based on the analysis elaborated in chapter 4.

#### **a) Game engines**

In this case, the best-known alternatives are considered, since they have a larger community. This is especially useful when it is the first time that an application is developed using a game engine. These game engines are Unreal Engine 4 from Epic Games and Unity3D 5 from Unity Technologies

It should be noted that the two alternatives are valid for this project, given that they are two consolidated development platforms in application development. After analyzing the main characteristics of each alternative, it has been decided to use Unity3D as a game engine. This decision has been taken because of its simplicity, in addition to the large community of users it has. Unity3D is the best choice for beginners while, using Unreal Engine, it is necessary to devote more time to learning. In addition, Unity has many free libraries that facilitate application development, and its cross-platform compatibility can be positive for future compatibility with other platforms.

#### **b) Devices**

One option would be HMDs without a built-in screen. The resolution of these devices is variable because it depends on the smartphone that is used as a screen. In addition, they do not provide controllers with forms of interaction that facilitate user's immersion. Also, it is the HMDs most likely to cause dizziness or cybersickness.

Given the limitations of HMDs without built-in screen and considering that the project is designed for a professional environment, in which general-purpose computers are usually used, this option is discarded. For the same reason, VR glasses designed for consoles are also discarded due to their limitations in terms of processing (although they have a built-in screen, such as Sony PlayStation VR). Therefore, there were considered those virtual reality glasses that support PC platforms.

It should be noted that an Oculus VR helmet was already available, specifically the Oculus Rift CV1 model. It is not necessary the purchase of another device, as the differences are not significant for the fulfilment of the project. However, HTC Vive Pro HMD would be the option chosen in the future in case of a decrease in its price. For this reason, Oculus Rift CV1 is the selected HMD.

### c) Assets

Among the different assets used, the most important are the following:

- **Virtual Reality Toolkit (VRTK).** It is the most complete asset to provide facilities when developing a VR project. It is free and compatible with Oculus SDK, and useful for interactions with the virtual world, such as the use ray casting or showing virtual hands in the position where controls are located, among others.
- **Online Maps.** This asset from Infinity Code allows map creation and interaction. This payment software was already owned beforehand and was used to experiment with map interaction and visualization. It is an option that fulfills its functions, and it was not considered the acquisition of another alternative, since Online Maps is valid for the accomplishment of the objectives of this project.
- **Photon Unity Networking (PUN).** It is the most used solution to create collaborative applications. The main reason why this alternative has been chosen is its extensive community, which has about 300,000 developers. It is characterized by its scalability and speed, in addition to including text and voice chat. Also, it offers cloud servers for application deployment. Another alternative was DarkRift Networking, but its community was significantly minor. However, it could be considered in the future, after having more experience with collaborative solutions.

## 12.3. Results

### 12.3.1. Software development project

For developing the application, a software development project was started. The development method followed is the cascade model with feedback. So, the project is divided into five phases:

- **Requirements analysis.** It is made the system requirements specification, in which its functionalities and restrictions are defined exhaustively.
- **Design.** The architecture of the system is detailed, making a division into different modules or components. The function of each component and the relationships between them are established.
- **Implementation.** Codification and construction of each module.
- **Testing.** It is verified the performance of the application.
- **Maintenance.** Error correction or performance improvement of software. del software after its delivery. This phase is not detailed in this document.

There is feedback between these phases, so that if an error is detected in any of them, it is possible to return to a previous stage to solve the problem.

### 12.3.2. General description of the developed application

It is a collaborative map visualization and interaction application. All users connect to the same room, in which a map will be placed. The room may contain a maximum of 8 users. The map is placed in front of the user as a vertical plane that shows the entire or a portion of its content, and which covers the entire field of vision of the user.

Each user can choose between a virtual reality environment with Oculus Rift HMD and Oculus Touch Controller or through a current monitor and an interaction with keyboard and mouse. Users can work jointly with the map, regardless of the chosen environment.

Users will have the ability to move the content of the map, as well as change its zoom level. Users always maintain a static position, in such a way that, both for the movement and for changing the zoom level, the content presented by the vertical plane is modified. Also, each user navigates independently through the map, so each of them observes the desired fraction of the map, without affecting other user's map.

Collaboration between users occurs using bookmarks. They can add, delete and highlight markers on the map. In virtual reality environments, a map position is indicated using the center of the gaze or a ray originating from the right hand, while the mouse cursor is used in non-VR environments. When a marker is indicated, it changes its color, suggesting that it is possible to perform an action on it, either remove or highlight it.

Highlighting a marker implies that it stands out regarding non-highlighted markers. Also, if a highlighted marker is outside the vertical plane, it will be displayed an arrow pointing to it. The color of the arrow will be the one associated with the user who highlighted the marker. To distinguish the actions of different users, each one will be associated with a different color, and it will be the color of their added and highlighted markers. Besides, a marker can only be highlighted by one user, as well as each user can only have one highlighted marker.

Additionally, the application allows the communication of the users of the room through a voice chat, in which each user can mute his microphone to stop transmitting audio.

In addition, it could be displayed a list of users who are in the same room, which could be hidden so as not to obstruct the map. This list of users shows, for each user, their alias, their associated color, if they are transmitting audio and if they have muted their microphone. In this way, it is easy to know who a marker belongs to or which user has highlighted a marker.

It should be noted that the use of markers is only related to pointing places on a map. In other words, this application does not calculate routes between markers nor show information relative to the marked places.

### 12.3.3. Testing results

A battery of tests was defined in chapter 8 to verify the correct functioning of the developed application. All the defined requirements, as well as the set of components, will be verified at the end of the tests.

After concluding the tests, some conclusions were drawn about it. It should be noted that most of the tests have been successful. However, slight errors have been found in successful tests, which will be considered in later versions of the application. Nevertheless, there were two unsuccessful test cases, which are mentioned by their identification code, as they were defined in chapter 8:

- **CP-29.** It was verified if the number of users in the room was limited to 8. The reasons why the ninth user was able to connect are unknown, since the maximum number of players was defined to 8 in codification phase. It is possible that this information must be considered by the developer in the implementation, and it is not Photon who is responsible for managing the number of connections. However, it is considered as a minor error since it does not influence on the completion of the objectives of the project.
- **CP-38.** It was the final test, in which the application had been tested with 8 simultaneous users for at least 10 minutes. A large performance loss was observed while using the Oculus Rift HMD. In this case, it could affect the completion of the objectives of the project.

The performance loss can be due to these causes:

- **Hardware limitations.** The processor or graphics card may have not supported the information processing of 8 users in a room and it affected on the performance of HMD. Therefore, it is recommended to test the application on a computer with more processing power.
- **Under-optimized code.** Due to the execution of many Update functions, in which you can perform a task in each frame, it was likely that frequent use of this type of functions when the number of users was high caused the performance loss. In later versions, the code will be analyzed in detail to find the most optimal solutions.
- **Overloaded HMD.** The tests were performed consecutively, requiring the continuous use of the VR device. Being the last test performed, it is possible that it was one of the causes of performance loss. Also, at the end of all tests, the HMD had a slightly higher temperature than usual. In case of this motive, it would be a temporary error that could be assumed.

## **12.4. Conclusions**

### **12.4.1. Achieved objectives**

At the end of the project, the objectives that were proposed at the beginning are analyzed, to check whether these have been accomplished or not.

The first objective of the project was to develop a VR collaborative map visualization and interaction environment. It was the main objective, so it was the first that was necessary to achieve. It is true that there were detected performance problems while executing a test case with eight simultaneous users, but this aspect does not affect the achievement of the proposed objective, since it had been ensured that the system worked correctly in scenarios with a smaller number of users. In addition, it cannot be affirmed with certainty that this performance loss is due to code inefficiency. Therefore, this objective has been accomplished, given that it has been possible to develop an application that satisfies the described characteristics.

Regarding the second objective, it was the extension of collaboration to environments without an immersion in a virtual reality environment. To achieve this, the application offers two types of environments that can collaborate with each other, and one of them does not require the use of a virtual reality HMD. In that case, the image is displayed through a conventional monitor. For this reason, this objective has also been accomplished.

The third objective was to take advantage of the implementation of a new technology to analyze new forms of interaction applicable to map visualization and interaction systems, which has also been achieved. To accomplish this objective, the VR environment allows two ways of indicating a position on the map. One of them is based on the use of the center of gaze, while the other creates a ray from the Oculus Touch Controller.

The simultaneous existence of two forms of interaction allows to compare them among themselves, in such a way that it will be possible to know which the most efficient alternative for this type of systems is.

After analyzing the objectives proposed at the beginning of the project, it can be concluded that these have been successfully achieved.

### **12.4.2. Future lines of work**

The present work serves as a starting point for applying virtual reality technology to map visualization and interaction environments. Therefore, future lines of work can be considered, either to offer other features or improve the existing ones.

Firstly, there could be considered the results obtained after the execution of test cases on the developed application, to perfect it. In this case, the priority is to detect the cause of the performance loss when reaching the maximum of eight simultaneous users. If the problem resided in the implementation, there would be adopted more efficient alternatives at a processing level.



After dealing with this issue, it would proceed to eliminate minor errors, such as hiding markers when the map offers its minimum zoom level, or fluidity problems when moving the map in the virtual reality environment.

Secondly, to verify the effectiveness of the forms of interaction implemented, there could be carried out experiments in which users with experience in map visualization and interaction environments would use the developed application. The objective of these experiments is to know what the most efficient alternative for this type of systems is.

Due to the limited time available for the development in this project, other forms of interaction that could be useful for the field of map visualization and interaction were not contemplated. Therefore, a possible line of work would be the addition of new forms of interaction based on leap motion technology, which allows interaction through hand movements. It is an alternative that favors immersion and the idea of presence.

Another line of work would be to use a curved plane to show the map, instead of a plain map, so that the distance between the user and all map points would be constant. With this, the user could observe all the elements of the map with the same size, which will allow the use of larger maps, as well as an improvement of user immersion.

Finally, it is proposed the creation of a configuration menu in which the user could customize the game controls, choosing the button associated with each of the available actions, as well as the possibility of creating multiple rooms to support multiple work teams.